İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

VERİ MADENCİLİĞİ SÜRECİNDE VERİ AYRIKLAŞTIRMA YÖNTEMLERİNİN KARŞILAŞTIRILMASI VE BİR UYGULAMA

Fatma Önay KOÇOĞLU
Enformatik Anabilim Dalı
Enformatik Yüksek Lisans Programı

Danışman
Yrd. Doç. Dr. Yalçın ÖZKAN
Temmuz, 2012

İSTANBUL

İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

VERİ MADENCİLİĞİ SÜRECİNDE VERİ AYRIKLAŞTIRMA YÖNTEMLERİNİN KARŞILAŞTIRILMASI VE BİR UYGULAMA

Fatma Önay KOÇOĞLU
Enformatik Anabilim Dalı
Enformatik Yüksek Lisans Programı

Danışman
Yrd. Doç. Dr. Yalçın ÖZKAN
Temmuz, 2012

İSTANBUL

2601090210 Öğrenci numaralı Fatma Önay KOÇOĞLU tarafından hazırlanan bu çalışma 02/07/2012 tarihinde aşağıdaki jüri tarafından Enformatik Anabilim Dalı Enformatik programında Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Jürisi

Yrd. Doç. Dr. Yalçın ÖZKAN (Danışman)
Zirve Üniversitesi
Mühendislik Fakültesi

Prof. Dr. Sevinç GÜLSEÇEN
İstanbul Üniversitesi
Enformatik Bölümü

Doç. Dr. Adem KARAHOCA
Bahçeşehir Üniversitesi
Mühendislik Fakültesi

Doç. Dr. Alp BARAY
İstanbul Üniversitesi
Mühendislik Fakültesi

Yrd. Doç. Dr. Çiğdem EROL
İstanbul Üniversitesi
Enformatik Bölümü

ÖNSÖZ

Yüksek lisans öğrenimim sırasında ve tez çalışmalarım boyunca engin bilgisi ile gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Yrd. Doç. Dr. Yalçın Özkan'a en içten dileklerimle teşekkür ederim.

Yıllarca, inatla, hayalinden hiç vazgeçmediğim akademisyenlik mesleğine başlamamda en büyük katkısı olan; her konuda desteğinden, anlayışından, sabrından birgün dahi mahrum kalmadığım; kelimeler ile anlatılamayacak kadar çok kıymetli hocam Prof. Dr. Sevinç Gülseçen'e teşekkür ederim.

Tez çalışmam süresince ve haricinde her türlü soru ve sorunlarıma bıkmak usanmak bilmeden, büyük bir sabırla çözüm üreten, yüzünde tebessümü hiç eksik olmayan, sevgili hocam Yrd. Doç. Dr. Çiğdem Erol'a teşekkür ederim.

Yüksek lisans öğrenimim boyunca her konuda destek olan, öğrendiğim birçok bilgide emeği olan değerli hocalarım Yrd. Doç. Dr. Zerrin Ayvaz Reis ve Yrd. Doç. Dr. Fatih Gürsul'a teşekkür ederim.

Tez çalışmamın büyük bir bölümünün yanı sıra özellikle uygulama aşamasında çok büyük desteği olan Öğr. Gör. Murat Gezer'e teşekkür ederim.

Bu çalışma boyunca iyi günde kötü günde desteklerini esirgemeyen beraber güldüğümüz, beraber ağladığımız, kalplerindeki sonsuz sevgi, yüzlerindeki tebessüm ile her durumda iyi hissetmemi sağlayan sevgili çalışma arkadaşlarım Arş. Gör. Elif Kartal Karataş'a ve Arş. Gör. Zeki Özen'e teşekkür ederim.

İstanbul Üniversite'li olduğum günden bugüne her türlü yardımı esirgemeyen Fen Fakültesi personeli Sadık Kaymak'a, Enformatik Bölümü ailesine ve tüm arkadaşlarıma teşekkür ederim.

Bugüne gelmemde en büyük paya sahip ömrümün en anlamlı varlıkları, yaratılırken bana lutfedilen en büyük şans olduğuna inandığım babam Mustafa Koçoğlu'na, annem Muhterem Koçoğlu'na ve kardeşim Elif Seda Koçoğlu'na emekleri, sonsuz sabırları, doğru ya da yanlış adımlarımda hiç şüphesiz gösterdikleri destek için teşekkür eder ve hazırlamış olduğum bu çalışmamı ithaf ederim.

Temmuz, 2012 **Fatma Önay KOÇOĞLU**

İÇİNDEKİLER

ŞEKİL LİSTESİ

TABLO LİSTESİ

SEMBOL LİSTESİ

α	: Anlamlılık düzeyi
A_{ij}	: i. aralık ve j. sınıftaki örneklerin sayısı
C_j	: j. sınıfın örnek sayısı
E_{ij}	: A_{ij}'nin beklenen frekansı
H	: Kesim noktası değeri
H(A)	: A'nın entropi değeri
H(A,B)	: A ve B'nin kazanç ölçütü
k	: Sınıf sayısı
m	: Sınıf sayısı
n	: Örnek sayısı
N	: İki aralıktaki toplam örnek sayısı
$p_{j,sağ}$	: Sağda yer alan örneğin j. sınıfa ait olma olasılığı
$p_{j,sol}$	: Solda yer alan örneğin j. sınıfa ait olma olasılığı
$p_{sağ}$	: Örneğin kesim noktasının sağ tarafında yer alma olasılığı
p_{sol}	: Örneğin kesim noktasının sol tarafında yer alma olasılığı
P	: Olasılık değeri
R_i	: i. aralığın örnek sayısı
t	: Aralık sayısı
v_{max}	: Aralığa ait maksimum değer
v_{min}	: Aralığa ait minimum değer
w	: Aralık genişliği
x	: Sürekli değişkenin sahip olduğu örnek sayısı

ÖZET

VERİ MADENCİLİĞİ SÜRECİNDE VERİ AYRIKLAŞTIRMA YÖNTEMLERİNİN KARŞILAŞTIRILMASI VE BİR UYGULAMA

Toplumlar, farklı ihtiyaçlar doğrultusunda çeşitli dönüşüm süreçlerinden geçmiş, bu süreç günümüzde bilgi merkezli hale gelmiştir. Ancak amaç bilgi yığınına değil doğru ve değerli bilgiye sahip olmaktır. Bu noktada ise veri madenciliği oldukça önem kazanmaktadır.

Veri Madenciliği, belirli yöntemlerin kullanılması ile var olan gizli bilgiyi ortaya çıkarma sürecidir (Özkan, 2008). Günümüzde her türlü veri, veri tabanları veya veri ambarlarında tutulmaktadır. Ancak tutulan tüm bu verilerin doğru olduğunu söylemek imkansızdır. Verilerin eksik ya da gerçeğe uygun olmayan yanlış şekilde girilmesi, aynı anlamdaki birden fazla verinin gereksiz var olması ve verilerin tutarsız olması veri madenciliği süreci sonrası elde edilecek bilgilerin yanlış ve doğrudan uzak olmasına neden olabilir. Verilerin etkin ve verimli bir şekilde işlenebilir ve yorumlanabilir olması için verilerin belirli kalite kriterlerini karşılayabilir olması gerekmektedir (Müller ve Freytag, 2003). Veri madenciliği farklı adımlardan oluşmak ile beraber bu adımlardan biri verilerin ön işlenmesidir. Nitelikli bilgilerin elde edilmesi nitelikli veriler ile sağlanabilir. Dolayısıyla bu adım elde edilecek sonuçlar için oldukça önemlidir.

Veri önişleme süreci içerisinde yer alan adımlardan birisi de veri ayrıklaştırmasıdır. Veri ayrıklaştırma işlemi için farklı yöntemler kullanılmaktadır. Bu yöntemlerden hangisinin daha etkin olduğu merak konusudur. Buradan yola çıkılarak bu tez çalışmasında veri kümelerine farklı ayrıklaştırma yöntemlerinin uygulanması ve hangi yöntemin daha etkin olduğu hususunun incelenmesi amaçlanmıştır.

Çalışma kapsamında Wisconsin Üniversitesi Hastaneleri'nde meme kanseri teşhisi sonucu ameliyat edilen hastalardan alınan örneklerin yer aldığı Wisconsin veri kümesi kullanılmıştır. Bu veri kümesi üzerine KEEL veri madenciliği yazılım aracı yardımı ile 1RD, CADD, CAIM, Chi2, ChiMerge, ID3, Eşit Genişlikli, Eşit Frekanslı olmak üzere sekiz farlı ayrıklaştırma yöntemi uygulanmıştır.

Chi2, ChiMerge, CAIM algoritmalarının gözle görülür bir oranda tutarlı çalıştıkları, 1RD algoritmasının genelde bir, ID3 algoritmasının da çok sayıda kategorik değişken elde ettiği belirlenmiştir. Bunun gibi çok sayıda kategorik değişken atayan farklı ayrıklaştırma yöntemlerinin, aynı nitelik değeri için birebir olmasa da çok yakın kategorik değişkenler atadığı belirlenmiştir.

Elde edilen bu sonuçlar veri madenciliği çalışmasında kategorik niteliklerle çalışmak gerektiğinde nasıl bir yol izlenmesi gerektiğini ve uygun yöntemin seçilmesi hususunda yol göstermektedir. Türkiye'de bu alanda yapılan tez çalışmaları taranmış olup literatürde eksikliği gözlemlendiğinden çalışmanın sonuçlarının literatüre de katkısının olması beklenmektedir.

SUMMARY

COMPARISON OF DATA DISCRETIZATION METHODS IN DATA MINING PROCESS AND AN APPLICATION

Societies have different needs in various conversion processes in the past but today this process has based on the information. The goal, not having a heap of information, is having accurate and valuable information. At this point, data mining is very important.

Data Mining is a process that reveals confidential information with using certain methods (Özkan, 2008). Today, all kind of data are kept in databases or data warehouses. However, it is impossible to say that all of this data is true. Missing or incorrectly entered data, having multiple redundant data that have same meanings, inconsistent data may cause obtaining incorrect information after data mining process. To interpret or to process the data effectively and efficiently, the data has to have certain quality criterias (Müller & Fraytag, 2003). The first step of data mining is preparing the data. Obtaining quality information can be provided with qualified data. Therefore, this step is very important for results obtained.

One of the step in the process of data preprocessing is data discretization. There are different methods used for data discretization process. It is an enigma that which of these methods is more effective. Thus, In this thesis study implementation of data discretization methods on the different data sets and to investigate which method is more efficient is aimed.

Within this study, to apply the selected data discretization methods wisconsin data set which cases from a study that was conducted at the University of Wisconsin Hospitals about patients who had undergone surgery for breast cancer, was selected. On this data set with the help of data mining software tool KEEL; as 1RD, CADD, CAIM, Chi2,

ChiMerge, ID3, Equal Width, Equal Frequency eight different discretization methods are applied.

Chi2, ChiMerge, CAIM algorithms work consistently in a considerable proportion, usually with 1RD algorithm one and with ID3 algorithm a large number of categorical variables were obtained. The discretization methods' that assigns too many categorical variable, categorical values are too close but not same.

These results lead that how to study when working with categorical attributes is needed and how to select the appropriate method. In addition, the lack of literature in this field in Turkey is scanned and results of the study are expected to contribute to the literature to fill a gap.

1.GİRİŞ

İçinde bulunduğumuz rekabetçi ortamda kurumların sahip olduğu veriyi yararlı bilgiye dönüştürerek bu rekabete ayak uydurabilmeleri ve avantajlı duruma geçmeleri önem taşımaktadır. Böyle bir amaca ulaşabilmek için veri madenciliği yöntemlerine gereksinim duyulmaktadır.

Veri madenciliği birçok adımdan oluşan bir süreç olarak karşımıza çıkmaktadır. Veri madenciliği yöntemlerinin uygulanması öncesinde, sağlıklı bir çözümleme yapılabilmesi için verinin bir ön işleme aşamasından geçirilmesi önem taşımaktadır. Aslında veri madenciliği projelerinin başarılı olması, veri ön işleme adımındaki çalışmaların başarısına bağlıdır.

Birleştirme, örnekleme, boyut indirgeme, özellik alt kümesi seçimi, özellik oluşturma, ayrıklaştırma, ikili hale getirme, değişken dönüşümü olmak üzere çok sayıda önişleme adımı bulunmaktadır. Bu tez çalışması kapsamında bu adımlardan biri olan veri ayrıklaştırma yöntemleri ele alınacaktır.

Veri madenciliği yöntemlerinin uygulanabilir olması, bazı durumlarda veri kümesindeki sayısal sürekli değerli niteliklerin kategorik hale çevrilmesini gerektirmektedir. Bu çevirme işi ayrıklaştırma olarak ifade edilmektedir.

Veri ayrıklaştırma işlemi için çok sayıda yöntem geliştirilmiştir. Bu tez çalışmasının amacı, en çok tercih edilen sekiz adet yöntemi ele alarak, bu yöntemler arasında bir karşılaştırma yapılması, hangilerinin daha etkin yöntemler olduğunun ortaya çıkartılması, bu yöntemlerin birbirlerine göre avantaj ve dezavantajlarının ortaya konmasıdır. Tezin özet bölümünde tez konusunun temel hatları ile bilgilendirmesi yapılmış, bu konunun seçilme nedeni ve önemi, tez kapsamında yapılan uygulamaya ve uygulamada kullanılan malzemelere ait kısa bilgiler ve elde edilen sonuçlar paylaşılmıştır.

Hazırlanan tezin ilk bölümü olan ‘Giriş’ kısmı yapılan çalışma hakkında bir öngörüş bildirme, tezin konusu hakkında kısa bir bilgilendirme yapma ve özet kısmında kısaca bahsedilen çalışmanın ana hatları, temel başlıkları ve bu başlıklara ait içerik açıklamalarını belirtmek üzere hazırlanmıştır.

Çalışmanın ‘Genel Kısımlar’ bölümünde çalışmanın alanı olan veri madenciliği süreci ile ilgili temel bilgiler yer almaktadır. Bu kapsamda veri madenciliği tanımı, bu süreçte kullanılan temel kavramlar, sürecin işleyişi, veri önişleme tanımı ile methodları ve bazı veri madenciliği yöntemleri ile bu yöntemlere ait örneklere yer verilmiştir.

Çalışmanın üçüncü bölümü olan ‘Veri Ayrıklaştırma Yöntemleri’, tez çalışmasının ana konusu olup ayrıklaştırma süreci, ayrıklaştırma yöntemleri hakkında temel bilgiler, ve algoritmalarından oluşmaktadır.

‘Malzeme ve Yöntem’ bölümünde uygulama çalışması ile ilgili olarak veri toplama yöntemi ve kullanılan veri kümesine ait özellikler, ayrıklaştırma yöntemlerinin uygulanması süreci ile ilgili adımlar ve bu süreçte kullanılan yazılım ile ilgili açıklamalara, uygulama yöntemi ve uygulama sürecinde yapılan işlemlere yer verilmiştir.

‘Bulgular’ bölümünde yapılan uygulama sonuçları paylaşılmış, sonuçlar ile ilgili açıklamalara yer verilmiştir. Buna göre bulgular “Kesim Noktası Sayılarına Göre Bulgular”, “Kategorik Değişkenlere Göre Bulgular” ve “Diğer Bulgular” olmak üzere üç başlık altında belirtilmiştir. “Kesim Noktası Sayılarına Göre Bulgular” başlığı altında algoritmaların veri kümesi üzerinde uygulanması sonrası oluşan aralıkların sayısını ifade eden kesim noktası sayıları ve varsa bu nokta değerlerinin listeleri yer almaktadır. “Kategorik Değişkenlere Ait Bulgular” başlığı altında da uygulanan ayrıklaştırma yöntemleri sonucunda veri kümesinde yer alan dokuz niteliğin değerlerine atanan kategorik değişkenlerin listesi ve yöntemlerin birbirleri ile olan tutarlılık yüzdeleri yer almaktadır. “Diğer Bulgular” başlığı altında ise kullanılan yazılımın ürettiği diğer sonuçlar paylaşılmıştır.

Tez çalışmasının ‘Tartışma ve Sonuç’ bölümünde bir önceki bölümde yer alan bulguların ve çalışma boyunca yazılım ve alan ile ilgili karşılaşılan eksikliklerin değerlendirilmesi, tezin amacı olan ayrıklaştırma yöntemlerinin hangilerinin aynı veri kümesi üzerinde çalıştırıldıkları taktirde benzer sonuçlar elde ettiği tartışılarak ve literatür taraması sonucu elde edilen avantaj ve dezavantaj değerlendirmeleri de göz önüne alınarak çıkarımlar yapılmıştır. Buna göre aynı ayrıklaştırma yöntemi kategorisine giren yöntemlerin benzer sonuçlar elde ettiği, farklı yöntemlerin avantajlarının yanı sıra aynı anda dezavantajlarının da olabileceği, her veri kümesine tüm ayrıklaştırma yöntemlerinin uygulanabilirliğinin yeni bir çalışma konusu olabileceği sonucuna ulaşılmıştır.

2. GENEL KISIMLAR

2.1. VERİ VE BİLGİ

Temel olarak veri; "*Olgu, kavram ya da komutların, iletişim, yorum ve işlem için elverişli, biçimsel ve uzlaşımsal bir gösterimi*", bilgi ise "*Bilgi işlemde, kullanılan uzlaşımsal kurallardan yararlanarak kişinin veriye yönelttiği anlam*" (Bilişim Terimleri Sözlüğü, BSTS, 1981 Akt:TDK) şeklinde ifade edilmektedir. Bu iki ifadeden genel olarak işlenmemiş bilgi veri; işlenmiş veri ise bilgidir sonucuna ulaşılabilir ve veri ile bilgi arasında çift taraflı gerçekleşen bir dönüşümün söz konusu olduğu söylenebilir.

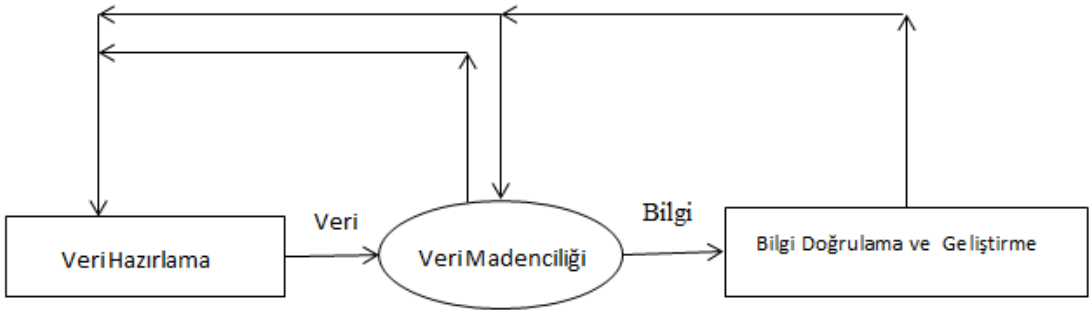
Yalçınkaya (2012), veri ve bilgi arasındaki bu dönüşümü veri-enformasyon-bilgi kavramları ile beraber bu kavramlar arasındaki ilişkiyi de ele alarak açıklamış, buna göre veriyi herhangi bir gerçek hakkındaki değeri tutan karakterler topluluğu, enformasyonu verinin derlenmesi ile beraber amaca yönelik olarak organizasyonu ve bilgiyi ise enformasyonu kullanarak getirilecek yeni bakış açılarının bütünü olarak tanımlamıştır.

Tanımlardan da anlaşılacağı üzere bilginin varlığı verinin varlığı ile mümkündür. Bu da bilgiye ulaşma çabası içerisinde her türlü verinin veri tabanlarında depolanması sonucuna götürmüştür. Ancak kısa sürede hızla artış gösteren, saklanan veri boyutunun büyüklüğü ve veri tabanlarında kullanılan sorgulama, raporlama araçlarının bilgiye ulaşma amacında yetersiz kaldığının görülmesi "Veri Tabanlarında Bilgi Keşfi" sürecinin ön plana çıkmasına neden olmuştur (Akpınar, 2000).

Frawley ve diğ. (1992) çalışmalarında, bilgi keşfini "*Veri içerisinde gizli, daha önceden bilinmeyen ve faydalı bilginin ortaya çıkartılması işidir*" şeklinde tanımlamışlardır. Ayrıntılı tanımlamalara sonraki kısımlarda yer verilecek olması ile beraber; Veri Madenciliğininin genel anlamda "gizli bilgiye ulaşma işi" olarak ifade edilmesi, bilgi keşfinin tanımı da göz önüne alındığında veri madenciliği ile bilgi keşfi işinin bazı

araştırmacılar tarafından eş değer kabul edildiği söylenebilir. Ancak Fayyad (1996), bilgi keşfi ve veri madenciliği tanımları eş değer kullanılıyor olsalar dahi bilgi keşfinin bilgi çıkarımı, fonksiyonel bağımlılık analizi, bilgi toplama, veri örüntüsü çıkarma işlemlerini içerdiğini; veri madenciliğinin ise herhangi ön bir tez olmadan, yeni bilgi keşfetmek için yapılan veri analizleri olarak tanımlandığını belirterek bu iki sürecin birbirinden farklı olduğunu savunmuştur.

Freitas (2002), veri madenciliğinin bilgi keşfi sürecinin bir alt adımı olduğunu ifade ederek Şekil 2.1'deki süreç tablosunu çalışmasında paylaşmıştır.



Şekil 2.1: Veri Keşfi Süreci (Freitas, 2002)

Kestirme, bilinen değişkenlerin değerlerinden, ilgilenilen bir diğer değişkene ait bilinmeyen bir değerin tahmin edilmesi işidir (Hand ve diğ., 2001). Bilgi keşfi sürecinde elde edilen bilgilerin değerlendirmesinin yapılarak ileriye yönelik kestirmelerin yapılması işi de veri madenciliği ile bilgi keşfini birbirinden ayıran adımlardan bir tanesidir.

2.2. VERİ MADENCİLİĞİ

Veri madenciliği günümüzde hızla gelişme gösteren bir alan olup, hem akademik hem de sektörel alandaki araştırmalarda uygulama alanı bulduğu gözlemlenmektedir.

"Veri madenciliği nedir?" sorusu, Friedman'ın (1997) çalışmasında yer alan Ferruzza, Zekulin, John ve Fayyad'a ait şu dört tanım ile cevaplanmaktadır.

- Ferruzza: Bilgi keşfi sürecinde veriler arasındaki daha önceden bilinmeyen ilişki ve örüntüleri ayırt etmek için kullanılan yöntemler bütünüdür.
- Zekulin: Büyük veri tabanlarından daha önceden bilinmeyen, kavranabilir, işlenebilir verileri ayıklama ve bu verileri önemli kararlar vermede kullanma sürecidir.
- Fayyad: Veri içerisindeki potansiyel olarak kullanışlı ve yararlı, anlaşılabilir örüntüleri belirleme sürecidir.
- John: Veri içersindeki avantajlı örüntüleri açığa çıkarma sürecidir.

Bu dört tanım incelendiğinde, ortak olarak veri madenciliğinin yararlı örüntüleri ortaya çıkarma süreci olarak ifade edildiği görülmekle beraber, özellikle örüntü kavramının kullanıldığı dikkat çekmektedir. Fayyad ve diğ. (1996), örüntü çıkarmayı, veri kümesinden yapı elde etme ve yüksek düzeyli veri alt kümesi ya da model oluşturma işi olarak tanımlamışlardır. Kısaca var olan büyük veri kümesinden anlamlı, işe yarar bir alt veri kümesi ya da kısaca bilgi elde etme işidir.

Veri madenciliğinin akademik ve sektörel bazda çok çeşitli kullanım alanı olmakla beraber, özellikle büyük şirketlerin stratejik plan geliştirme ve karar verme süreçlerinde etkin bir şekilde kullanıldığı söylenebilir. Veri madenciliğinin kullanıldığı diğer alanlara örnek olarak müşteri kayıplarının nedenlerinin bulunması, sepet analizlerinin yapılması ve buna bağlı olarak pazarlama stratejilerinin geliştirilmesi, eğilim analizlerinin yapılması, dolandırıcılıkların ortaya çıkarılması ve hastalık nedenlerinin araştırılması sayılabilir.

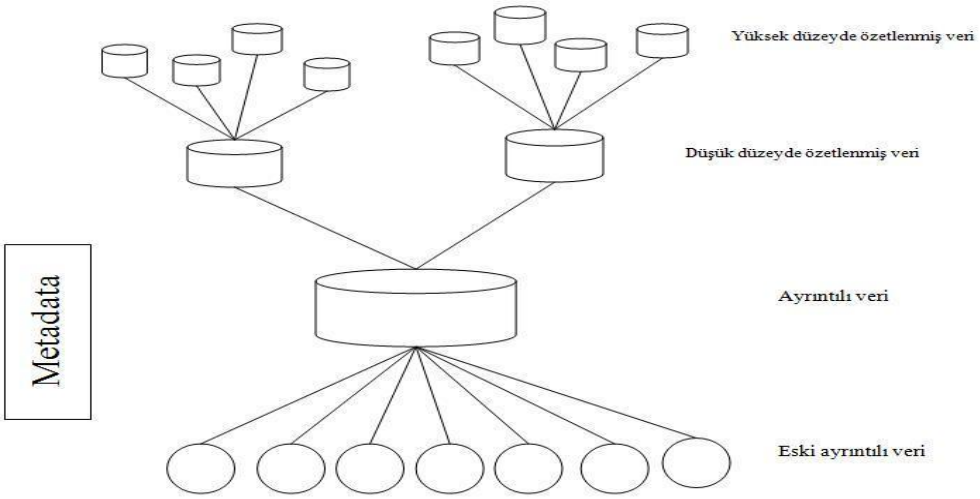
2.2.1. Veri Madenciliğinde Kullanılan Bazı Temel Kavramlar ve Veri Ambarı

Kurumların ürettiği veri günümüzde gelişmiş veri tabanı sistemleri ve veri ambarlarında saklanmaktadır. Veri tabanı sistemleri veri kümelerinin düzenli bir şekilde tutulduğu ve bu veri kümeleri üzerinde yönetebilme, güncelleyebilme gibi işlemlere olanak sunan ortamlar olarak tanımlanabilir. Veri ambarları da yine verilerin tutulduğu ortamlardır. Ancak veri tabanı ile arasında bazı farklılıklar söz konusudur. Veri ambarları veri tabanlarının yetersiz geldiği durumlarda veriye hızlı erişilerek verinin işlenmesini sağlayan sistemlerdir.

Inmon (2000) veri ambarını, "*karar verme sürecinini destekleyen, konuya yönelik, bütünleşik, zaman boyutu olan, kalıcı veri topluluğu*" olarak tanımlamıştır. Burada konu tabanlı olması belirli bir konuya yönelik olmasını, bütünleşik olması tutarlı adlandırma, kodlama, fiziksel özellik vb. gibi formatlarda verilerin düzenlenebilmesini, zaman boyutu olması geçmiş dönemlere ait verinin de sistemde yer alacağını, kalıcı olması ise güncellemeye izin verilmeyeceğini ifade etmektedir.

William H. Inmon ve Chuck Kelley veri ambarlarının sahip olması gerektiği özellikleri belirlemişlerdir. Bunlardan en önemlisi veri ambarının; metadata, ayrıntılı veri, eski ayrıntılı veri, düşük düzeyde özetlenmiş veri ve yüksek düzeyde özetlenmiş veri olmak üzere beş düzeyde veriye sahip olacağıdır (Inmon ve Kelly, 1994). Şekil 2.2'de bu beş düzey verinin gösterimi yer almaktadır.

- Metadata: Kullanılan verinin yapısını, konumunu, düşük ve yüksek seviyede özetlenme bilgilerini tutan veridir.
- Ayrıntılı veri: En son olayları içeren verilerdir.
- Eski ayrıntılı veri: Daha eski tarihli ayrıntılı verilerdir.
- Düşük düzeyde özetlenmiş veri: Ayrıntılı veriden süzülerek elde edilen veridir.
- Yüksek düzeyde özetlenmiş veri: Ayrıntılı veriden süzülmüş olup kolayca erişilebilir veridir.



Şekil 2.2: Veri ambarında yer alan verilerin yapısı (Inmon, 2000)

Veri ambarının özel bir mimari yapısı vardır. Veri ambarları işlevsel kaynak sistemi, veri depo alanı, veri sunum alanı ve veri erişim aracı olmak üzere dört temel yapıdan meydana gelmektedir (Kimball ve Ross, 2002). Veri ambarı mimari özelliklerine bakıldığında şu üç önemli aşamanın gerçekleştiği söylenebilir:

a. Kaynaktan alınan verinin dönüştürülmesi: Farklı veri tabanlarında farklı biçimlerde tutulan verinin aynı biçime dönüştürülmesi sağlanır.

b. Veri ambarının oluşturulması: Dönüştürülen veriler veri ambarına aktarılır ve veri ambarına ait veri tabanı oluşturulur.

c. Kullanıcıların veri ambarına erişebilmeleri: Oluşturulan veri ambarına farklı sorgular ile kullanıcıların erişebildiği aşamadır.

Özetlenirse, temel olarak veri ambarları, işlenmek üzere belirli özelliklere sahip çok büyük veri yığınlarının tutulduğu yer şeklinde de ifade edilebilir. Bu büyük veri yığını içerisinden değerli bilgiyi keşfetme işi veri madenciliğidir. Veri ambarı içerisinde hatalı veriler bulunabilir. Veri içinde tutarsız, eksik veri olabileceği gibi hatalı veri de bulunabilir. Veri toplama ya da giriş işlemlerinde yapılan hatalar sonucu elde edilen hatalı veriye de veri madenciliğinde "gürültülü" adı verilir. Tüm bu eksik, tutarsız,

gürültülü veriden kurtulmak ve veri madenciliği yöntemlerini uygulamak için takip edilecek belirli bir süreç söz konusudur.

2.2.2. Veri Madenciliği Süreci

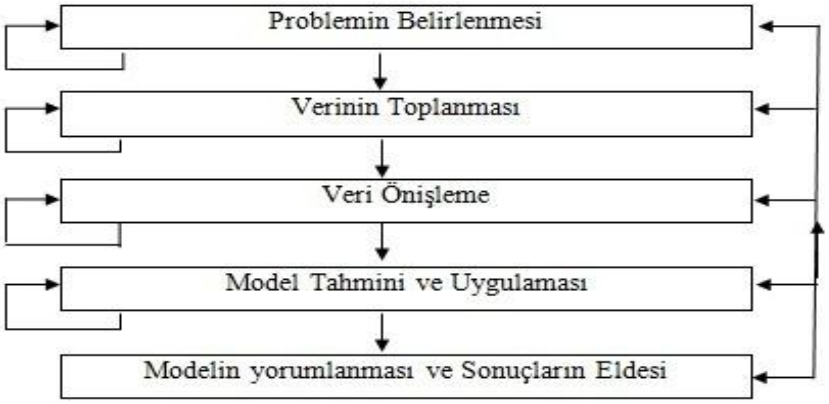
Genel olarak bakıldığında, veri madenciliği işinin veri modelleme ve bilginin ortaya çıkarılması biçiminde olduğu görülebilir. Modelleme, veriyi oluşturan nitelikleri ve ilişkileri anlama (Gunn, 1997), bağımlılıkları belirleme (Kavi, 1993) ve bunların sonucu olarak mevcut veriye dayalı sonuçların üretilmesi; belirlenen kural ve sonuçların ortaya çıkan yeni gözlemlere uygulanması işlemidir.

Veri madenciliği süresince izlenecek adımları Han ve Kamber (2001) daha ayrıntılı biçimde ele almış ve çalışmalarında şu şekilde sıralamıştır:

a. Veri temizleme: Gürültülü, eksik veya uygun olmayan veriler için temizleme işlemi gerçekleştirilir. Temizleme aşamasında şu yöntemlerden herhangi biri kullanılabilir (Sullivan, 2011; Roldan, 2010):
   - Eksik veya yinelenen verilerin atılması
   - Eksik değerler için sabit bir değer tanımlanması
   - Eksik değer yerine diğer verilerden istatistiksel yollarla elde edilecek diğer bir verinin girilmesi
   - En çok kullanılan ortak bir veri girilmesi
   - Maksimum ve minumum değerlere uyumlu olması için normalizasyon işlemlerinin yapılması.
b. Veri bütünleştirme: Farklı biçimdeki veriyi aynı biçime getirme işlemidir.
c. Veri seçimi: Sonucu değiştirmeyeceği inanılıyorsa kullanılacak veri sayısını azaltıp çıkan sonucu daha sonrasında genelleştirme işlemidir. Bir bakıma örnekleme de denilebilir.
d. Veri dönüştürme: Söz konusu verilerin analiz için uygun hale getirilmesi yani normalleştirilip standartlaştırılması işlemidir.
e. Veri medenciliğin yöntemlerinin veriye uygulanması: Yukarıda sıralanan işlemler yapılıp veri uygun hale getirildiğinde veri madenciliği algoritmalarından probleme uygun olanı uygulanır.

f. Elde edilen sonuçların sunumu: Uygulanan veri madenciliği algoritmalarının sonuçları elde edildikten sonra ilgili birimlere uygun şekilde sunumunun yapılması işlemidir.

Han ve Kamber'in belirlediği sürecin yanı sıra Kantardzic (2011), veri madenciliği sürecini daha genel adımlar ile ifade ederek; bu süreci Şekil 2.3'te görüldüğü biçimde problemin belirlenmesi, verinin toplanması, verinin önişlenmesi, modelin tahmini (belirlenmesi) ve veri madenciliği yönteminin uygulanması, modelin yorumlanması ve sonuçların elde edilmesi şeklinde beş adımda ifade etmiştir.



Şekil 2.3: Kantardzic'e göre (2011) veri madenciliği süreci

2.2.3. Veri Önişleme

Han ve Kamber'in çalışmasının yanı sıra Kantardzic'in çalışmasından da görüldüğü üzere veri madenciliği sürecinde iki adım önemli bir yer teşkil etmektedir. Bunlardan ilki veri önişlenmesi diğeri ise modelin elde edilmesi ve veri madenciliği yöntemlerinin uygulanmasıdır.

Han ve Kamber (2001), veri önişleme adımının önemini veri madenciliği sürecinin doğruluk ve etkinliğinin bu adımdaki işlemler sonucu elde edilecek kaliteli veriler ile gerçekleştirilebileceği şeklinde ifade etmektedir. Veri önişleme süreci temel olarak veri temizleme, bütünleştirme ve yapılandırma olmak üzere üç aşamadan meydana gelmektedir (Tanasa ve Trousse, 2004) .

Tan ve diğ. (2006), veri önişleme adımının birçok farklı stratejik ve teknik yöntem içerdiğini belirtmiş bunun yanı sıra en önde gelen teknikleri yedi başlık altında toplamıştır. Bu başlıklar şu şekilde özetlenebilir:

a. Birleştirme: Toplanan verilerin bir özet halinde ifade edilme biçimidir. Örneğin satış miktarlarının şehirsel yerine bölgesel olarak veya yağış miktarının aylık yerine yıllık olarak belirtilerek verilerin bölgesel ve yıllık değişkenleri altında birleştirilmesi işlemidir.
b. Örnekleme: Ana veri kümesinden bu veri kümesinin özelliklerine çok yakın başka bir deyişle kümeyi en iyi şekilde temsil edebilecek bir alt veri kümesi seçilmesi işlemidir. Gerçekleştirilecek işlemler bu alt veri kümesi üzerinde uygulanacak, elde edilen sonuçlar da genelleştirilecektir.
c. Boyut indirgeme: Bazı veri kümeleri fazla sayıda nitelik içermektedir. Böyle durumdaki veri kümeleri fazla boyutlu olarak değerlendirilir. Bu durum görselleştirmenin zorlaşmasına, bellek ve zaman kaybına neden olmaktadır. Boyut indirgeme, tüm bu olumsuzlukların engellenmesi için veri kümesinin boyutunun düşürülmesi işlemidir.
d. Nitelik alt kümesi seçimi: Bazı veri madenciliği uygulamalarında çok sayıda nitelik modele dahil edilmektedir. Bu niteliklerden bazıları analizlerin sonucunu etkileyemeyecek türdendir. Analizleri kolaylaştırmak ve modellerin karmaşıklığını azaltmak için söz konusu niteliklerin belirlenerek modelden çıkarılması gerekir. Bunun için nitelik alt kümesi seçme yöntemleri uygulanır.
e. Nitelik oluşturma: Veri kümesinde yer alan niteliklerden önemli ve daha etkin yeni niteliklerin oluşturulması işlemidir.
f. Ayrıklaştırma ve ikili hale getirme: Bazı veri madenciliği yöntemleri, özellikle kümeleme algoritmaları sürekli nitelikler yerine kategorik niteliklere, bazıları da sürekli veya kategorik nitelikler yerine bir ya da daha fazla ikili niteliğe ihtiyaç duymaktadır. Kategorik niteliklere dönüştürme işlemine “ayrıklaştırma”, ikili niteliklere dönüştürme işlemine de “ikili hale getirme” denir.
g. Değişken dönüşümü: Her nesne için o nesneye ait değişken değerinin dönüştürülmesi işlemidir. Bir değerin sadece büyüklüğü önemli ise bu değişkene ait değerlerin mutlak değerinin alınarak dönüştürülmesi bu yönteme örnektir.

Yukarıda sözü edilen yedi veri önişleme adımından “ayrıklaştırma” işlem adımı bu çalışmanın ana konusunu teşkil ettiğinden ilerleyen bölümlerde bu adıma ait ayrıntılar ele alınacaktır.

Veri ön işleme adımlarının, var olan veri kümesini kalite kriterlerine uygun hale getirmek için kullanıldığı göz önüne alınarak sözü edilen kalite kriterleri Müller ve Freytag (2003) tarafından hiyerarşik olarak şu şekilde sıralanmıştır:

a. Doğruluk: Gerçek veri kümesinin; herhangi bir anormallik, bütünlük ve geçerliliği bozacak veri içermeden, bu veri kümesine ait varlıkların tümünü birebir temsil etmesi.

    Bütünlük: Geçersiz ve bütünlüğü bozacak değerlerin olmamasıdır.

    Tutarlılık: Verilerin arasında uyumluluğun sağlanıp tutarsızlığın kaldırılmasıdır.

    Yoğunluk (Çözünürlük/Hassasiyet): Verilerin doğru olmasıdır. Ancak değeri olmayan ve null değer ile belirtilen değerler için tahmin yapmak veri kalitesini düşürebileceğinden bu değerlerin null değer içermesi kalite kriterleri açısından bir sorun teşkil etmemektedir. Dolayısıyla tahmin etmek yerine null değerler tercih edilebilir.

b. Benzersizlik: Herhangi bir veri tekrarına yer vermemektir.

Veri önişleme adımının ardından yukarıda sıralanan kriterlere uygun veriye veri madenciliği yöntemlerinin uygulanması söz konusu olur.

2.2.4. Bazı Veri Madenciliği Yöntemleri

Veri Madenciliği için birçok yöntem bulunmaktadır. Bu yöntemler farklı avantaj ve dezavantajlara sahiptirler. Bu avantajlar ve dezavantajlar veri kümelerinin büyüklüğüne, veri kümeleri içerisinde yer alan örneklerin türlerine ve analizlerdeki belirli hedeflere bağlıdır (Shmueli ve diğ., 2010).

Gorunescu (2011), veri madenciliği yöntemlerinin çözüm sağlayabileceği problemleri sınıflandırma, kümeleme, birliktelik kuralları belirleme, ardışık örüntü keşfi, regresyon analizleri ve sapma denetimleri olmak üzere altı farklı başlık altında gruplandırmıştır.

Dunham (2003), sınıflandırmayı verileri daha önceden tanımlanmış grup ya da 'sınıf'lar ile eşleştirme olarak tanımlamış, bu alandaki en yaygın örnek olarak kredi risklerinin değerlendirmesi olarak ifade etmiştir. Sınıflandırma, veri tabanındaki verinin kullanılarak kuralların belirlenmesi ve bu kurallara uygun olarak da yaşanabilecek benzer durumlarda nasıl karar alınması gerektiğine dair karar yapısı oluşturulması mantığına dayanmaktadır. En yakın k-komşu algoritması, ID3, CART gibi karar ağacı algoritmaları bu kategoride sayılabilecek örneklerdir.

"Kümeleme", verilerin benzer özelliklerinin belirlenerek gruplandırılmasıdır. Gruplandırma işlemi kavramsal kümeleme ilkelerinden sınıflararası maksimum ve sınıf içi minimum benzerlik ilkelerine dayalı olarak gerçekleştirilir (Chen ve diğ., 1996). Kapsama ağacı algoritması (Spanning Tree Algorithm), K-Means algoritması bu kategoride örnek verilebilecek veri madenciliği algoritmalarıdır.

"Birliktelik kuralları", veri tabanında var olan verileri ilişkilendirerek eş zamanlı gerçekleşme olasılıklarının hesaplanması olarak da ifade edilmektedir. Çoğu kişinin bildiği sepet analizi örneği olan bir babanın bira ile bebek bezini aynı anda alması olayından hareketle alışverişin yapıldığı süpermarket tarafından sonuçlar çıkartılması ve uygun reyonların dizayn edilmesi durumu bu kategoriye örnek olarak verilebilir. Apriori algoritması bu kategoride örnek verilebilecek veri madenciliği algoritmasıdır.

"Ardışık örüntü keşfi", belirli bir zaman aralığında gerçekleşen olayın bağlantılı olduğu olayların ortaya çıkartılması ya da gerçekleşen olaylar ile bir bağının olup olmadığının araştırılıp, olaylar örüntüsü kurabilmek olarak kısaca ifade edilebilir. Belirli bir sürede hastanın takip edilerek bu süre içerisinde görülen belirtilerin birbirleri olan ilişkisini belirlemek bu problem için verilebilecek örneklerden bir tanesidir.

"Regresyon" iki veya daha fazla değişken arasındaki ilişkinin ortaya çıkartılmasıdır. Bir şirketin pazarlama reklamları ve satış rakamları arasında ilişkinin ortaya çıkartılması bu

analize örnek olarak gösterilebilir. Veri madenciliğinde regresyon analizi yöntemlerinden de yararlanılmaktadır.

"Sapma analizleri" veya "anormalliklerin belirlenmesi" olarak da ifade edilebilen bu analizde olağan dışında davranışlar belirleyen verilerin tespiti için veri madenciliği yöntemlerinden yararlanılabilmektedir. Örneğin Takçı ve Soğukpınar (2002) çalışmalarında, güvenlik açığı nedeni ile meydana gelebilecek saldırı tespiti probleminde iki temel anormallik yaklaşımı belirleyerek k-komşu algoritmasının bu problemde kullanılabileceğinden bahsetmişlerdir.

Çok sayıda veri madenciliği yöntemi bulunmakla beraber aşağıda en çok kullanılan veri madenciliği algoritmalarından bazıları açıklanmaya çalışılmıştır.

2.2.4.1. En Yakın K-Komşu Algoritması

En yakın k-komşu algoritması sınıflandırma yöntemlerinden birisidir. Amaç, sınıfları belirli olan örnek kümeye ait gözlemlere yeni bir gözlem eklendiğinde bu gözlemin hangi sınıfa ait olacağının belirlenmesidir. Tanımdan da anlaşılacağı üzere geçmiş verilerin varlığına dayalı olup, gürültülü verinin varlığında sorun yaratır (Han ve Kamber, 2001).

Bu algoritma sınıflandırma işlemi için gözlemler arası uzaklık hesabına dayanır. Son eklenen gözleme var olan gözlemlerin uzaklıklarının hesaplanması ve en az uzaklığı sahip k tane gözlemin seçilmesi işlemi gerçekleştirilir. Uzaklık hesabı için Öklid uzaklık formülü kullanılır (Bao ve diğ., 2005).

2.2.4.2. Genetik Algoritmalar

Genetik algoritma evrimsel açıdan bilinen kalıtım sürecine benzer şekilde işlemektedir. Amaç çok boyutlu uzayda aramak ve en iyinin var olmasını sağlamaktır. Bu algoritmada uygunluk fonksiyonu oluşturularak en yüksek uygunluk fonksiyonu değerine sahip bireylerin, uygunluk fonksiyon değeri yüksek olan diğer bireyler ile bir araya gelip yeni bireyler elde edilmesi sağlanır. Bu şekilde ne kadar çok birey bir araya getirilir ise çalışma o kadar iyi anlamına gelmektedir.

Genetik algoritmaların iş, bilimsel ve mühendislik alanları olmak üzere birçok farklı kullanım alanları bulunmaktadır. Sınıflandırma ve bilgi tabanlı sistemleri tasarımı, sinir ağı mimarilerinin otomatik belirlenmesi, görüntü işleme ve örüntü sınıflandırma metodolojileri geliştirilmesi genetik algoritmaların uygulamalarına örnek olarak verilebilir (Bandyopadhyay ve diğ., 2007).

2.2.4.3. Apriori Algoritması

Apriori algoritması en çok bilinen birliktelik kuralı algoritmasıdır. Veri kümesinde yer alan bir öğenin ele alınarak kendisinden sonra gelen ve aday olarak belirlenen öğeler ile karşılaştırılması ve aralarındaki ilişkinin ortaya çıkartılmasını ve bu karşılaştırma sonrası bazı aday öğelerin atılması ile sonuca ulaşılmasını sağlayan algoritmadır (Liang ve diğ., 2010).

Bu algoritma genelde satışlar sonrası satışı yapılan ürünler arasında satış ilişkisinin ortaya çıkartılması ile örneklenmektedir. Kimi zaman bir markette kimi zaman kozmetik dükkanında satılan malzemelerin hangileri bir arada satılıyor sorusunun cevabı eldeki veri kümeleri üzerine bu algoritmanın uygulanması ile cevap bulmaktadır. Burada yapılan işlem satışı yapılan ürünlerin öncelikle satış miktarlarının belirlenmesi, sonrasında ikili ve üçlü gruplar halinde algoritmaya uygun şekilde destek sayılarının hesaplanarak elemelerin gerçekleştirilmesi veya gruplama işlemine devam edilerek satılan ürünlere ait güven sayılarının da yardımı ile birliktelik kurallarının çıkarımının yapılması olarak ifade edilebilir.

Bir mağazada satılan ürünler için bu algoritmanın uygulandığı varsayıldığında izlenmesi gereken adımlar şu şekildedir:

- Eşik değerler belirlenir.
- Her bir ürün için veri kümesi taranır, o ürünün kaç defa tekrar ettiği saptanarak destek sayısı belirlenmiş olur.
- Bundan sonraki her öğe için veri kümesi tarandığında bu destek sayısının eşik değerinden büyük olması beklenir. Aksi durumda bu öğe elenir.
- Tüm öğeler tarandıktan sonra sürekli arttırılmak koşulu ile ikili, üçlü vb. gruplandırmalar ile taramalar gerçekleştirilir.
- Elde edilen destek ve güven ölçülerine göre birliktelik kuralları oluşturulur.

Bilinmesi gereken destek ve güven ölçütleri aşağıdaki şekilde açıklanmaktadır.

destek(A): A olayının gerçekleşme olasılığı:

$$\text{destek}(A)=\text{sayı}(A) / \text{Toplam olay sayısı} \quad (2.1)$$

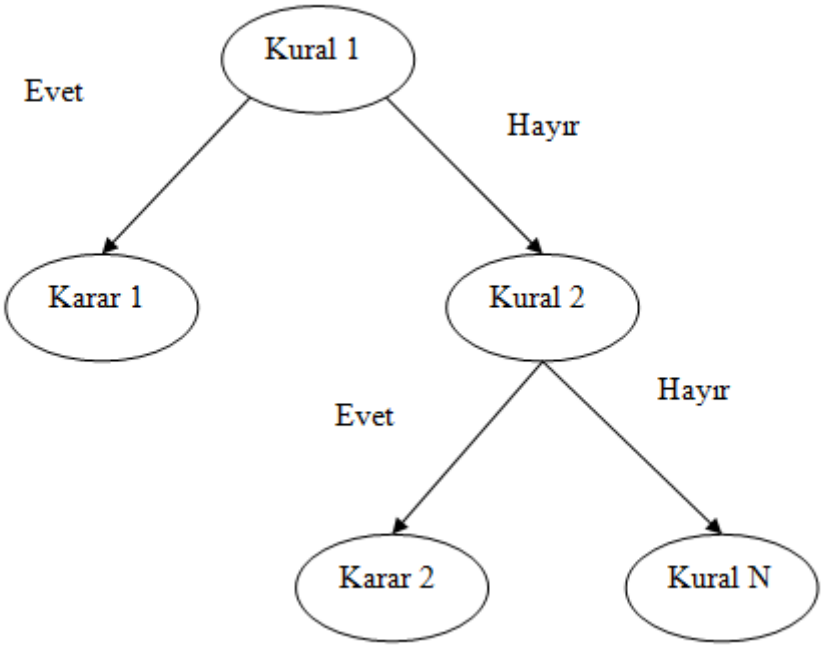
güven(A→B): A olayı ile beraber B olayının da gerçekleşme olasılığı:

$$\text{güven}(A \rightarrow B)=\text{sayı}(A,B)/\text{sayı}(A) \quad (2.2)$$

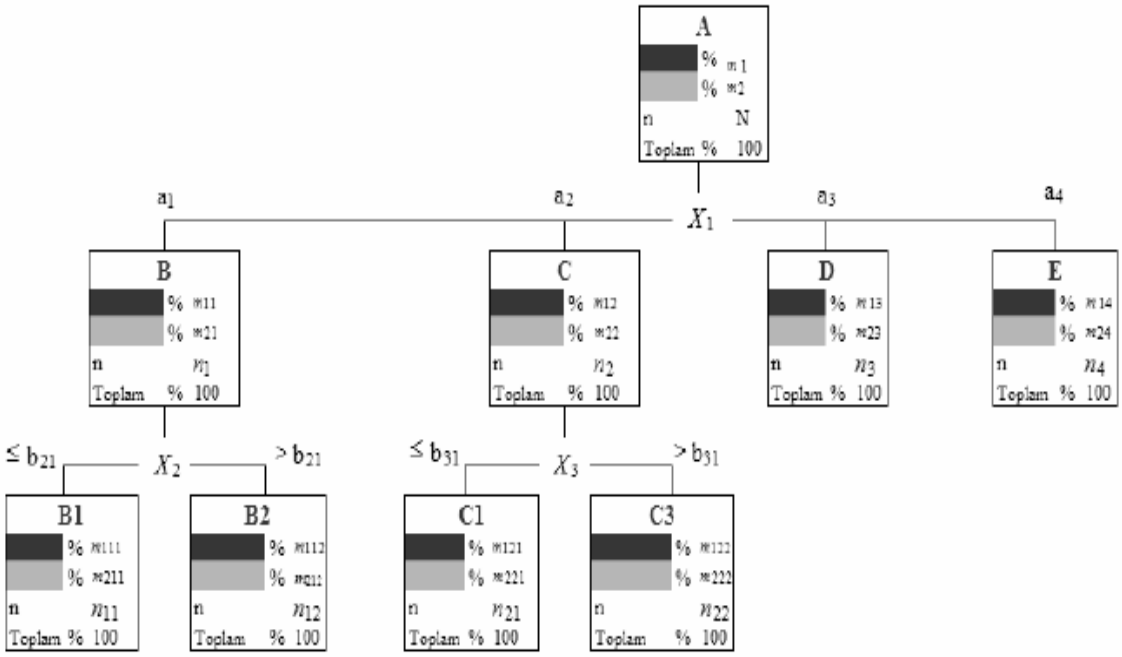
2.2.4.4. Karar Ağacı Algoritmaları

Karar ağacı algoritmaları, sınıflandırma yöntemine ait algoritmalar arasındadır. Tahmin etme ve tanımlamaya yönelik karar ağaçları veri tabanı sistemleri ile uyumlu, maliyetinin az ve kurulumunun kolay olması nedeniyle veri madenciliğinde tercih edilen yöntemlerden birisidir. Karar ağaçlarını diğer yöntemlerden üstün tutan özelliği, çıkarılan kuralların daha anlaşılır olması ve kesin sonuçlar elde edilebilmesidir (Mitchell, 1997).

Karar ağacı algoritmasında ağaç şeklinde sınıflandırıcılar oluşturulmaktadır. Karar ağaçları kök, yaprak ve dallardan oluşur. Bir ağaç nasıl ki kökünden başlayıp dallanıp yapraklarına doğru gidiyorsa bahsedilen ağaç veri yapısı da bir kök düğüm, buna bağlı dallar ve en sonunda da yaprak adı verilen öğelerden meydana gelen bir yapıdır. Kök düğüm en üst yapı olmak üzere, yaprak düğüm en son yani kendisinden sonra herhangi bir düğüm gelmeyen düğümlerdir. Basit bir karar ağacı yapısı Şekil 2.4'te verilmiştir (Türkoğlu, 2007). Daha kapsamlı bir karar ağacı ise Şekil 2.5'te yer almaktadır (Koyuncugil, 2007).



Şekil 2.4: Basit bir karar ağacı yapısı (Türkoğlu, 2007)



Şekil 2.5: Kapsamlı bir karar ağacı yapısı (Koyuncugil, 2007)

Aşağıda karar ağaçlarının açıklanmasını kolaylaştıran ve yaygın bir şekilde kullanılan hava problemi örneği yer almaktadır. Bu örnek Quinlan (1993) tarafından ID3 algoritması yardımıyla oluşturulmuştur (Akt: Özkan, 2008).

Tablo 2.1: Quinlan (1993) problemi için veri kümesi

Hava	**Isı**	**Nem**	**Rüzgar**	**Oyun(Tenis)**
Güneşli	Sıcak	Yüksek	Hafif	Hayır
Güneşli	Sıcak	Yüksek	Kuvvetli	Hayır
Bulutlu	Sıcak	Yüksek	Hafif	Evet
Yağmurlu	Ilık	Yüksek	Hafif	Evet
Yağmurlu	Soğuk	Normal	Hafif	Evet
Yağmurlu	Soğuk	Normal	Kuvvetli	Hayır
Bulutlu	Soğuk	Normal	Kuvvetli	Evet
Güneşli	Ilık	Yüksek	Hafif	Hayır
Güneşli	Soğuk	Normal	Hafif	Evet
Yağmurlu	Ilık	Normal	Hafif	Evet
Güneşli	Ilık	Normal	Kuvvetli	Evet
Bulutlu	Ilık	Yüksek	Kuvvetli	Evet
Bulutlu	Sıcak	Normal	Hafif	Evet
Yağmurlu	Ilık	Yüksek	Kuvvetli	Hayır

Tablo 2.1'deki niteliklerden oyun niteliği hedef belirlenen niteliktir. Çünkü farklı durumlara göre oyun oynayıp oynanamayacağı kararı verilecektir. Oyun niteliği için iki değer vardır; evet ve hayır. Bu iki değere ait olasılık dağılımları aşağıdaki gibidir.

$$P_{Oyun} = (5/14, 9/14) \tag{2.3}$$

Oyun niteliği için entropi değeri ise,

$$H(Oyun)=-\Sigma p_i \log_2(p_i)= -(5/14 \log_2 5/14 + 9/14 \log_2 9/14)$$

$$= 0.940$$

şeklinde hesaplanır.

Bu aşamadan sonra dallanmaların nasıl olması gerektiğine karar verilmesi gerekmektedir. Bu adımda diğer niteliklerin kazanç ölçütleri aşağıdaki formüle göre hesaplanır:

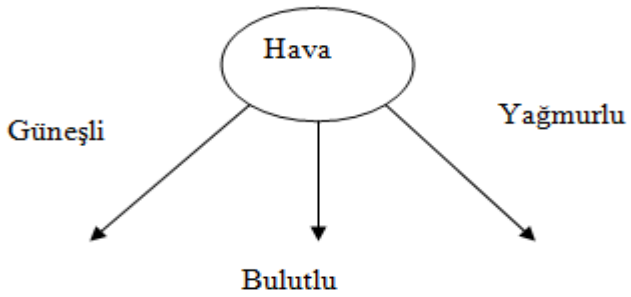
$$H(X, T)= \Sigma |T_i /T| H(T_i) \quad (2.4)$$

Bu formüle göre tüm kazançlar tek tek hesaplandığında ortaya çıkan ilk basamak kazanç miktarları Tablo 2.2'de yer almaktadır.

Tablo 2.2: Quinlan (1993) problemi için ilk basamak kazanç tablosu

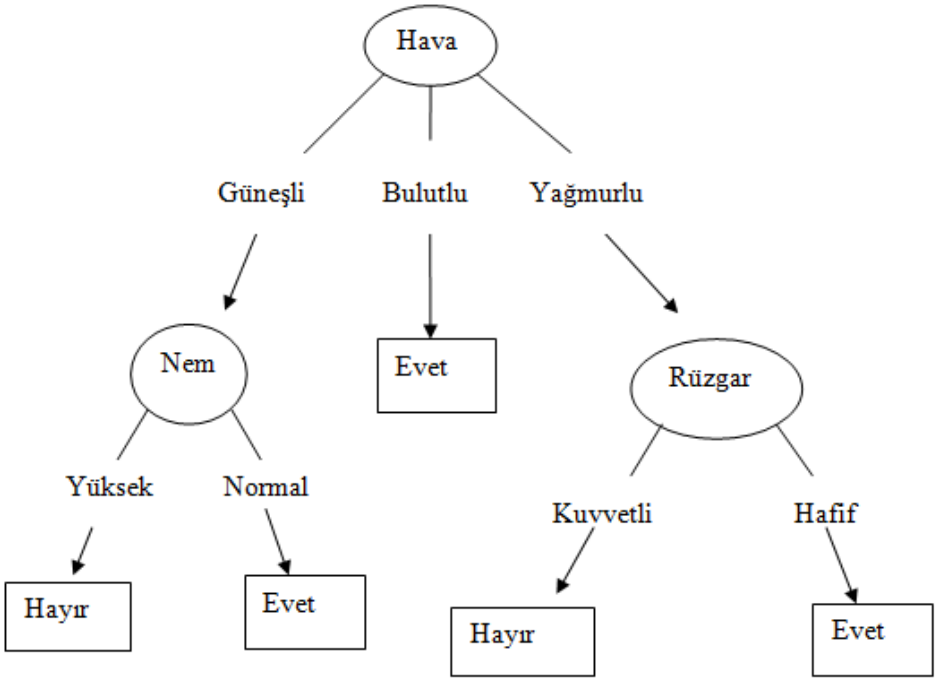
Nitelik	Kazanç
Hava	0.246
Isı	0.029
Nem	0.151
Rüzgar	1.048

Tablo 2.2'deki değerlere göre en fazla kazanç hava niteliğinden olduğundan başlangıç düğümü Hava olarak seçilir. Havaya ait üç gözlem değeri olduğundan üç dallanma söz konusu olur. Şekil 2.6'da Hava niteliğine ait dallanmalar yer almaktadır.



Şekil 2.6: Hava Niteliği İle İlgili Dallanmalar

Bu aşamadan sonra her alt nitelik için (Güneşli, Bulutlu, Yağmurlu) dallanmaların nasıl olacağı belirlenir. Bunun için de yine yukarıda verilen formüller kullanılarak kazanç ölçütleri ayrı ayrı belirlenir. Bu problemin sonucunda Şekil 2.7'deki karar ağacı elde edilir.



Şekil 2.7: Quinlan (1993) problemi için üretilen karar ağacı

3. VERİ AYRIKLAŞTIRMA YÖNTEMLERİ

3.1. VERİ AYRIKLAŞTIRMA

Uygulamada karşılaşılan veri sürekli veya kesikli (ayrık) veri biçiminde olabilmektedir. Özellikle tahmin modelleri geliştirme çalışmalarında kesikli veri sürekli veriye tercih edilmektedir. Bu tercih nedenleri arasında kesikli verilerin bilgi düzeyli gösterimli olması, bazı işlemler sonrası sadeleştirilmiş olması, anlaşılır ve açıklanabilir olması sayılabilir (Olson ve Delen, 2008). Sürekli verinin kesikli veriye dönüştürülmesi için bazı işlemler uygulanmaktadır. Bu işlemler "veri ayrıklaştırma" olarak bilinmektedir.

Veri ayrıklaştırma, veri ön işleme adımlarından birisidir. Sürekli verinin kesikli veriye dönüştürülmesi olarak kısaca ifade edilse de literatürde veri ayrıklaştırma için farklı şekillerde ifade edilmiş ancak aynı anlamı taşıyan birçok tanım yer almaktadır; Jin ve diğ. (2007), veri ayrıklaştırmayı en az veri kaybı ile sürekli verinin sonlu komşu aralıklar şekline dönüştürülmesi süreci olarak tanımlarken, Das ve Vyas (2010), aynı veya yakın özellikler taşıyan sürekli verinin grup veya aralıklara dönüştürülmesi süreci olarak ifade etmektedir. Yaş niteliğine ait değerlerin veri ayrıklaştırma sonrası elde edilecek aralıklara göre dağıtılıp bu aralıkların "genç", "orta yaş" ve "yaşlı" gibi üç kategori şeklinde ifade edilmesi veri ayrıklaştırma işlemine verilebilecek en basit örneklerdendir.

Veri ayrıklaştırma işlemi sonrası sürekli veri azaltılıp veri kümesi en iyi temsil edilecek şekilde özetlenirken, veri madenciliği sonrası ise elde edilen bilgilerin kullanımı ve sunumu kolaylaşmış, bunun yanı sıra bu bilgilerin daha anlamlı hale gelmesi de sağlanmış olur (Chakrabarti ve diğ., 2008).

Veri ayrıklaştırma işlemi için birçok yöntem kullanılmakla beraber birçok araştırmacı varolan yöntemlerin yanı sıra yöntemler üzerine yeni özellikler eklemek suretiyle farklı ayrıklaştırma yöntemleri geliştirmiştir. Bu çalışma kapsamında; en çok kullanılan ve

farklı yöntemleri temel alan yöntemler olmasına dikkat edilerek aşağıdaki veri ayrıklaştırma yöntemleri seçilmiştir. Bu yöntemler:

- ChiMerge Ayrıklaştırma Yöntemi
- Chi2 Ayrıklaştırma Yöntemi
- Eşit Genişlikli Ayrıklaştırma Yöntemi
- Eşit Frekanslı Ayrıklaştırma Yöntemi
- 1RD Ayrıklaştırma Yöntemi
- ID3 Ayrıklaştırma Yöntemi
- CADD Ayrıklaştırma Yöntemi
- CAIM Ayrıklaştırma Yöntemi

3.2. VERİ AYRIKLAŞTIRMA YÖNTEMLERİNİN SINIFLANDIRILMASI

Veri ayrıklaştırma yöntemleri özelliklerine göre belirli gruplar altında toplanabilmektedir. Dougherty ve diğ. (1995), bu gruplandırmayı denetimli ve denetimsiz, dinamik ve statik, yerel ve genel olmak üzere üç şekilde yaparken, Yang ve Webb (2005), çalışmalarında veri ayrıklaştırma yöntemlerini aşağıda belirtildiği şekilde dokuz grupta sınıflandırmıştır.

a. Denetimli ve Denetimsiz: Kesim noktalarını elde etmek için sınıf bilgisini kullanan yöntemler denetimli, kullanmayanlar ise denetimsiz yöntemler olarak ifade edilmektedir. Eşit genişlikli, eşit frekanslı ayrıklaştırma yöntemleri sınıf bilgisini kullanmadığından denetimsiz; ID3, 1RD gibi yöntemler sınıf bilgisini kullanmaları nedeniyle denetimli yöntemler sınıfına dahil olmaktadır.

b. Parametrik ve Parametrik olmayan: Her ayrıklaştırma aşaması için kullanıcıdan bir parametre girmesini bekleyen yöntemler parametrik, beklemeyen yöntemler ise parametrik olmayan yöntemler olarak ifade edilir.

c. Hiyerarşik ve Hiyerarşik olmayan: Hiyerarşik yöntemler kesim noktası seçimi yaparken artırımlı bir süreç izleyerek hiyerarşik bir yapı oluşturur. Bölümlemeli

ve birleştirmeli bir süreç izleyen yöntemler bu grup altında yer almakta olup ChiMerge, Chi2 bu hiyerarşik yöntemlere örnek gösterilebilmektedir.

d. Tekli ve Çoklu değişkenli: Çok değişkenli yöntemler ayrıklaştırma sürecinde nitelikler arası ilişkiyi dikkate alırken, tek değişkenli yöntemler herhangi bir nitelik değerine referans vermeden işlem yapar.

e. Ayrık ve Ayrık olmayan: Ayrık methodlar değer aralıklarını birbiri ile çakışmayan ayrık aralıklara böler.

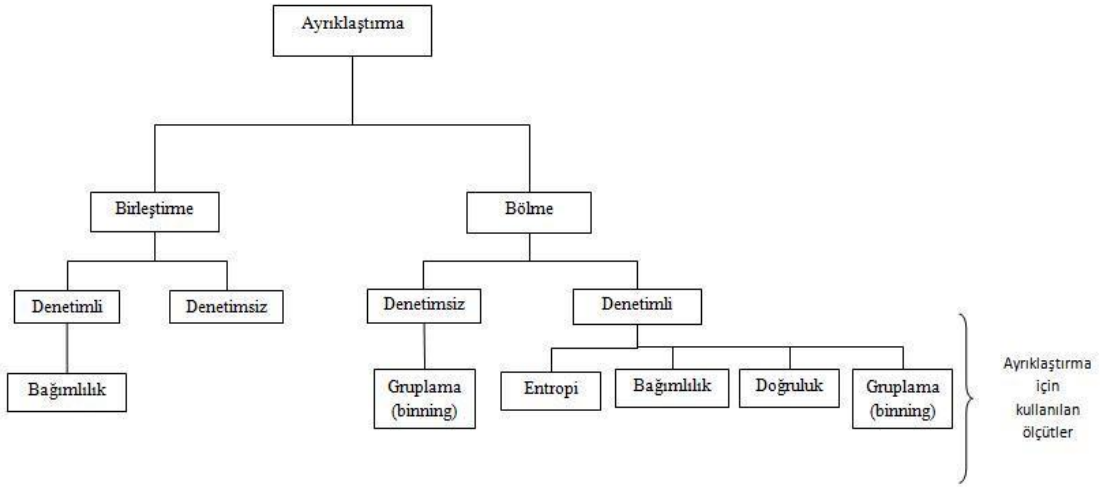
f. Genel ve Yerel: Genel ayrıklaştırma yöntemleri sadece bir sınıflandırma işi için eşleme fonksiyonu yaratırken yerel yöntemlerde bunun tam tersi durum söz konusudur. Yerel yöntemler örnek uzayın yerel bölümlerine uygulanmak üzere bölümler yaratırken, genel yöntemler n boyutlu örnek uzay üzerinde her bir niteliğin diğer niteliklerden bağımsız olarak farklı bir bölmeye gireceği bir ağ oluştururlar (Dougherty, 1995).

g. Gayretli ve Tembel: Gayretli yöntemler sınıflandırma süreci için öncelikli bir eşleme fonksiyonu oluştururken, tembel yöntemler sadece ihtiyaç duyulan durumlarda bunu yapar. Bunun yanı sıra tembel yöntemler her sürekli nitelik için bir test veri kümesi verilene kadar işlem yapmadan beklerken her bir değişkenin çevresinde sadece iki kesim noktası meydana getirir (Hsu ve diğ., 2003).

h. Sıralı ve Sembolik: Sıralı yöntemler nicelden nitel sıralı niteliklere bir eşleme fonksiyonu oluştururlar.

i. Bulanık ve Bulanık olmayan: Bulanık yöntemler bulanık eşleme oluştururken bu yöntemde değerler farklı genişliklerdeki birden fazla aralığa ait olabilir. Bunun yanı sıra bulanık ayrıklaştırma, sürekli nitelik değerindeki küçük değişimlerin bu niteliğe ait olasılıklar üzerindeki etkisinin küçük olması düşüncesine dayanır (Lee, 2005).

Yang ve Webb'in bu ayrıklaştırma yöntemleri gruplandırmasını Liu ve diğ. (2002) farklı bir çatı oluşturarak hiyerarşik bir düzende sunmuşlardır. Buna göre birleştirme ve bölme özelliğini temel almışlar bunun bir alt adımı olarak da denetimli ve denetimsiz yöntemler ayrımına gitmişlerdir (Şekil 3.1).



Şekil 3.1: Ayrıklaştırma yöntemleri için hiyerarşik bir çatı (Liu ve diğ., 2002)

3.3. VERİ AYRIKLAŞTIRMADA KULLANILAN BAZI YÖNTEMLER

Kabul görmüş, oldukça sık kullanılan veri ayrıklaştırma yöntemlerinden bazıları aşağıda açıklanmıştır.

3.3.1. ChiMerge

Randy Kerber tarafından geliştirilen ChiMerge; sınıf bilgisini kullanmasından ötürü denetimli, sınıf bağımlılıkların istatistiksel ölçünün baz alınarak belirlendiği (Maimon ve Rokach, 2005) ve buna göre ayrıklaştırma işleminin gerçekleştirildiği yöntemdir. χ^2 istatistiksel metodu ile komşu aralıklardan gelen sınıfların bağımlılıkları belirlenmekte, bağımsız olmaları durumunda da bu sınıflar birleştirilmektedir (Cerquides ve Lopez De Mantarase, 1997).

Kerber (1992) çalışmasında, bu yöntemi sonlandırma kriterine kadar devam edecek aşağıdan yukarıya komşu aralıkların birleştirilmesi işlemi olarak ifade ederek, tekrar eden iki temel adımdan meydana geldiğini belirtmiştir. Bu adımlardan ilki her komşu aralık çifti için χ^2 değerinin hesaplanması ve ikincisi ise tüm aralık çiftlerine ait χ^2 değerlerinin belirlenen χ^2 eşik değerinden (tavsiye edilen eşik değerleri:0.90, 0.95 veya 0.99) fazla oluncaya kadar birleştirilmesidir.

χ^2 değerinin hesaplamak için kullanılacak formül aşağıdaki gibidir:

$$\chi^2 = \sum_{i=1}^{2} \sum_{j=1}^{k} \frac{(A_{ij} - E_{ij})^2}{E_{ij}} \tag{3.1}$$

Burada k, sınıfların sayısıdır.

$$R_i = \sum_{j=1}^{k} A_{ij} \tag{3.2}$$

$$C_j = \sum_{i=1}^{2} A_{ij} \tag{3.3}$$

$$E_{ij} = \frac{R_i C_j}{N} \tag{3.4}$$

$$N = \sum_{i=1}^{2} R_i \tag{3.5}$$

R_i, i. aralığın örnek sayısı

A_{ij}, i. aralık ve j. sınıftaki örneklerin sayısı

C_j, j. sınıfın örnek sayısı

E_{ij}, A_{ij}'nin beklenen frekansı

N, İki aralıktaki toplam örnek sayısı

ChiMerge yöntemi algoritması kısaca şu şekilde ifade edilebilir:

a. Verilen nitelikler için artan düzende veriler sıralanır.
b. Bu veriler ve sınıflarına göre aralıklar belirlenir.

c. Belirlenen aralık çiftleri için χ^2 değeri hesaplanır.
d. Her aralık çifti için hesaplanan χ^2 değeri belirlenen eşik değerinin altında ise bu aralık çiftinin birleştirme işlemi gerçekleştirilir.
e. Her birleştirme sonrası oluşan yeni aralıklar için yeniden hesaplanan χ^2 değerleri eşik değerinin üzerinde oluncaya kadar birleştirme işlemi aynı şekilde devam ettirilir.

Literatürde yöntemi açıklamak için Tablo 3.1'de verilen örnek yaygın bir biçimde kullanılmaktadır.

Tablo 3.1: ChiMerge uygulaması için örnek veri kümesi

Örnek	**Nitelik Değeri**	**Sınıf**
1	1	1
2	3	2
3	7	1
4	8	1
5	9	1
6	11	2
7	23	2
8	37	1
9	39	2
10	45	1
11	46	1
12	59	1

Verilen nitelik değerleri için aralık kesim noktaları 0, 2, 5 ,7.5, 8.5, 10, …'dur.
[7.5, 8.5] ve [8.5, 10] aralık çiftleri için χ^2 değeri minimumdur.

Tablo 3.2: ChiMerge uygulaması için örnek veri kümesine ait aralıklar tablosu

	K=1	K=2	Toplam
Aralık [7.5, 8.5]	$A_{11}=1$	$A_{12}=0$	$R_1=1$
Aralık [8.5, 10]	$A_{21}=1$	$A_{22}=0$	$R_2=1$
Toplam	$C_1=2$	$C_2=0$	N=2

K nitelik ve sınıfları, A tek bir aralıkta yer alan tek bir niteliğe ait değerlerin sayısını, C belirlenen aralık çiftinde yer alan tek bir niteliğe ait değerlerin sayısını, R tek bir aralıkta yer alan ve tüm niteliklere ait değerler sayısını, N ise belirlenen aralık çiftinde yer alan ve tüm niteliklere ait değerlerin sayısını belirtmek üzere Tablo 3.2'teki veriler

yardımıyla (3.1), (3.2), (3.3), (3.4) ve (3.5)'teki formüller hesaplanırsa aşağıdaki sonuçlar elde edilir:

$E_{11}=1$, $E_{12}\approx 0.1$ ve $E_{21}=1$, $E_{22}\approx 0.1$

$\chi^2=0.2 < 2.706$ olduğundan bu iki aralık birleştirilir.

3.3.2. Chi2

Huan Liu ve Rudy Setino tarafından geliştirilen bir diğer istatistiksel bazlı ayrıklaştırma yöntemi Chi2'dir. ChiMerge algoritmasında olduğu gibi bu yöntemde de ayrıklaştıma işlemi için χ^2 yöntemi kullanılır (Liu ve diğ., 2002).

Liu ve Setiono (1995), çalışmalarında Chi2 ayrıklaştırma algoritmasının başlangıcında anlamlılık düzeyinin 0.5 gibi yüksek bir değer olarak belirlendiği, her bir niteliğin değerine göre sıralandığı, her örneğin kendisine ait tek bir nitelik değeri içeren aralığa eklendiğini belirtmişler ve iki adımdan meydana gelen algoritmayı şu şekilde açıklamışlardır:

Birinci adımda, her bir komşu aralık için denklem (3.5)'e göre χ^2 değeri hesaplanarak, bulunan en küçük χ^2 değerine göre aralıkların birleştirilmesi gerçekleştirilir. Oluşturulan tüm aralıkların χ^2 değeri anlamlılık düzeyi olarak ifade edilen parametreden fazla oluncaya kadar birleştirme işlemine devam edilir. Belirtilen bu işlemler, ayrıklaştırılmış veri içerisinde elde edilen tutarsız değerin, azalan bir anlamlılık düzeyi katsayısından fazla olması durumuna kadar devam ettirilir.

İkinci adımda her bir i. niteliğe ait anlamlılık düzeyi değerinin birleştirme işlemine yön vermesi açısından önemli olduğu göz önüne alınır. Niteliklerin birleştirilmesi işleminden sonra her seferinde tutarsızlık kontrolü yapılır. Tutarsızlık değerinin aşması durumunda, i. niteliğin anlamlılık düzeyi değeri, bu niteliğin bir sonraki birleştirme işlemi için azaltılır, aksi durumda i. nitelik bir sonraki birleştirme işleminde yer almaz. Hiçbir nitelik birleştirilemez hale gelene kadar ikinci adımın işlemleri tekrarlanır

Chi2 yönteminin ilk aşaması ChiMerge algoritmasının genelleştirilmiş hali olarak kabul edileceği gibi ChiMerge yönteminde var olan önceden tanımlanan anlamlılık düzeyi yerine Chi2 yöntemi, otomatik arttırılan bir eşik değerini, başka bir ifade ile azalan

anlamlılık düzeyi değerini kullanmaktadır (Tay ve Shen, 2002). Chi2 yöntemine ait algoritma aşağıda yer almaktadır.

Aşama 1: (att - nitelik)

```
set α  = .5;
do while (InConCheck(data) < δ){
        for each numeric att {
        Sort(att, data); /* nitelik üzerinden verileri sırala */
        chi-sq-init(att, data); /*veriyi yenile */
        do {
                chi-sq-calculation(att, data)
         } while (Merge(data))
       }
        α0 = α;
        α  =  decreSigLevel(α);
 }
```

Aşama 2:

```
set all sigLvl[i] = α0 for att i;
do until no-att-can-be-merged {
        for each mergeable att i {
        Sort(att, data); /* nitelik üzerinden verileri sırala */
        chi-sq-init(att, data); /* veriyi yenile */
        do {
                chi-sq-calculation(att, data)
        } while (Merge(data))
        if (InConCheck(data) < δ)
        sigLvl[i] = decreSigLevel(sigLvl[i]);
        else att i is not mergeable;
        }
}
```

3.3.3. Eşit Genişlikli Ayrıklaştırma

En yalın ayrıklaştırma yöntemlerinden biri olan Eşit Genişlikli Ayrıklaştırma Yönteminde bir değişken için gözlemlenen değerler için maksimum ve minimum değerler tanımlanır (Kurgan ve Cios, 2004). Bu değerler, eşit genişlikte olan ve kullanıcı tarafından belirlenmiş t tane aralığa yerleştirilir (Dougherty ve diğ., 1995). Herhangi bir sınıf bilgisi kullanmaması nedeniyle denetimsiz ayrıklaştırma yöntemleri arasında yerini almaktadır.

Bu yöntemin yalın olması diğer yöntemler arasında bu yöntemi daha cazip hale getirse (Boulle, 2005) de, aralık genişliklerinin belirlenerek eşit aralıklara bölme işlemi doğru yapılamadığı taktirde bu yöntem ayrıklaştırma işlemi sonrasında büyük bir veri kaybına neden olabilir (Wong ve Chiu, 1987).

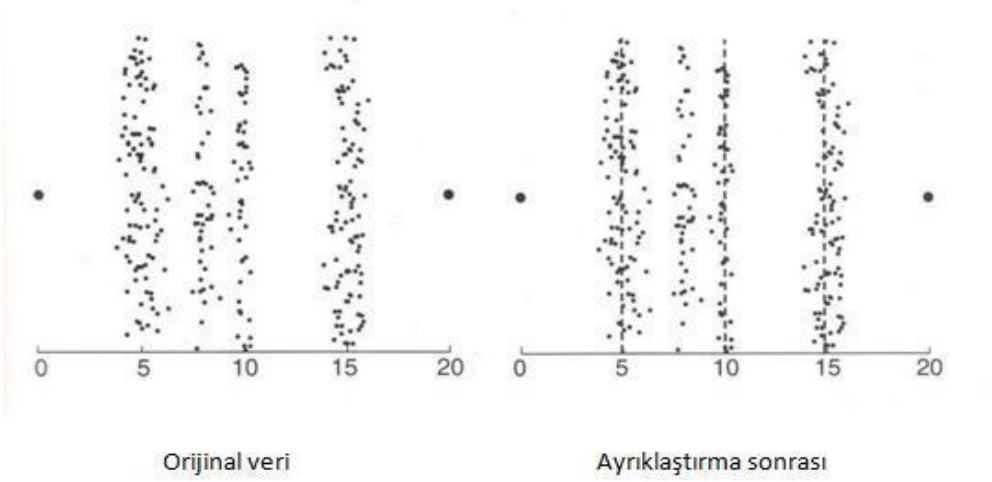
Yang ve Webb (2002), çalışmalarında bu yöntemi ve sonrasında oluşacak aralıkların kesim noktası olarak ifade edilen sınır noktalarını şu şekilde tanımlamışlardır:

$$w = (v_{maks} - v_{min}) / t \tag{3.6}$$

$$v_{min} + w,\ v_{min} + 2w,\ \ldots,\ v_{min} + (t-1)w \tag{3.7}$$

Bu formülleri kısaca açıklamak gerekirse, (3.6)'da maksimum (v_{maks}) ve minimun (v_{min}) değerler arasındaki farkın belirlenen aralık sayısına (t) bölünmesi ile elde edilen w değişkeni aralıkların genişliklerini belirlemektedir. (3.7)'deki minumum değerden başlamak üzere genişlik değerinin ve katlarının eklenmesi ile elde edilen değerler de aralıkların bitiminde yer alan ve kesim noktası olarak ifade edilen değerlerin büyüklüklerini ifade etmektedir.

Şekil 3.2'de orijinal veri ile eşit genişlikli ayrıklaştırma yönteminin uygulanması sonucu elde edilen verilerin görünümü yer almaktadır.



Şekil 3.2: Eşit genişlikli ayrıklaştırma yöntemi sonrası verilerin görünümü (Tan ve diğ., 2006)

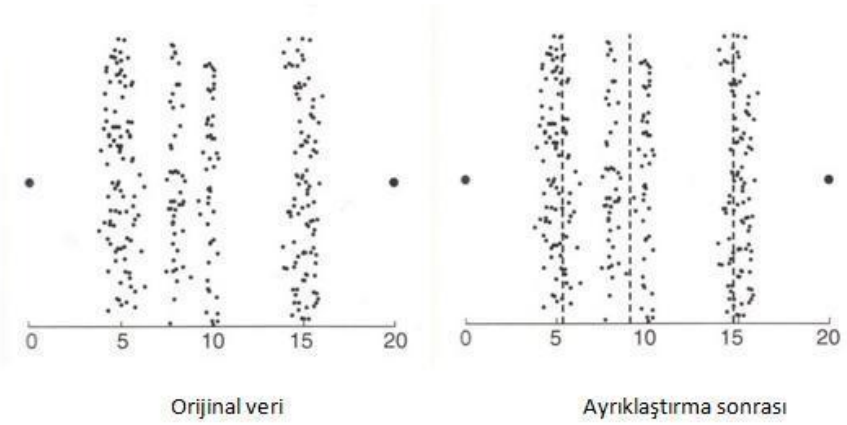
3.3.4. Eşit Frekanslı Ayrıklaştırma

Eşit frekanslı ayrıklaştırma yöntemi de eşit genişlikli ayrıklaştırma yöntemi gibi herhangi bir sınıf bilgisi kullanmadan uygulanan denetimsiz ayrıklaştırma yöntemlerinden birisidir (Biba ve diğ., 2007).

Bu yöntemde öncelikle veri kümesinde yer alan her bir hedef niteliğin bağlantılı olduğu diğer nitelikler tanımlanarak elde edilen bağlantılı nitelikler artan bir sıraya göre dizilmektedir. Bu işlemin ardından bu nitelik değerleri her bir aralığa eşit sayıda düşecek şekilde aralıklara yerleştirilmektedir (Mehta ve diğ., 2005; Kotsiantis ve Kanellopoulos, 2006).

Dougherty ve diğ. (1995), çalışmalarında bu yöntemin işleyişini biraz daha formülize ederek t tane aralığa, her aralıkta x/t tane olmak üzere eşit sayıda nitelik değeri gelecek şekilde yerleştirilme işlemi olarak bu yöntemi belirlemişlerdir. Burada x, sürekli değişkenin sahip olduğu örnek sayısını temsil etmektedir. Bu işlem t-1 aralığın oluşmasını sağlarken eşit genişlikli ayrıklaştırmada da olduğu gibi, her aralığa ait sınır değerler, bir başka deyişle kesim noktaları meydana gelmektedir. Komşu kesim noktaları arası alan normalize edilmiş histogramda 1/t değerine mümkün olabildiğince yakın olacaktır (Khan ve Revett, 2004).

Şekil 3.3'de orijinal veri ile eşit frekanslı ayrıklaştırma yönteminin uygulanması sonucu elde edilen verilerin görünümü yer almaktadır.



Şekil 3.3: Eşit frekanslı ayrıklaştırma yöntemi sonrası verilerin görünümü (Tan ve diğ., 2006)

3.3.5. ID3

Iterative Dichotomiser 3 (ID3) algoritması, John Ross Quinlan tarafından geliştirilmiş entropi ölçüsü kullanan karar ağacı oluşturma algoritmasıdır. Ancak bu algoritma kesikli ve kategorik değişkenler için geliştirilmiş olup, sürekli nitelikler algoritma sonunda meydana gelecek karar ağaçlarının boyutunun büyümesine neden olmaktadır (Ching ve diğ., 1995). Bu durumda, bir düğüm için çok sayıda dallanma oluşumunu engellemek için sürekli niteliklerin ayrıklaştırılması gerekmektedir (Liu ve diğ., 2002). Böylece bu yöntem ile karar ağacı oluşturulurken sürekli verinin ayrıklaştırılması işlemi de aynı anda gerçekleştirilmiş olur. Anlaşılacağı üzere ID3 ayrıca bir ayrıklaştırma algoritması değildir.

En yüksek bilgi kazancı ile beraber ayrıştırıcı niteliklerin seçilmesi bu yöntemin temel taşıdır (Chen ve Cheng, 2008). ID3 algoritmasında, var olan örnekleri kendi sınıflarına en iyi şekilde ayıracak niteliğin seçimi için ise sezgisel olarak entropi tabanlı bir ölçü kullanılır (Chang ve diğ., 2007). Kesim noktalarını bulmak üzere entropi tabanlı ölçü baz alınarak oluşturulmuş formül kullanılır (Liu ve diğ. ,2002):

$$H = -p_{sol} \sum_{j=1}^{m} p_{j,sol} \log p_{j,sol} - p_{sağ} \sum_{j=1}^{m} p_{j,sağ} \log p_{j,sağ} \quad (3.8)$$

m, sınıfların sayısı

p_{sol} ve $p_{sağ}$, örneğin kesim noktasının sırası ile sol veya sağ tarafında yer alma olasılığı

$p_{j,sol}$ ve $p_{j,sağ}$, sırası ile sol ve sağda yer alan örneğin j. sınıfa ait olma olasılığı

3.3.6. 1RD

1 Kural Ayrıklaştırma (1-Rule Discretization:1RD) algoritması, Robert Holte tarafından 1993 yılında geliştirilen, sınıf bilgisi kullanan dolayısı ile denetimli ayrıklaştırma yöntemlerinden birisidir. Holte (1993) çalışmasında; kullanılacak veri kümesinin bir parçasına ait örneklerin alınıp bunun sonucu olarak gerçek veri kümesi için 1-kural elde edilmesini sağlayan sistemden bahsetmiş, bunun yanı sıra programın tüm sayısal nitelik değerlerini sürekli nitelik olarak algılaması nedeniyle bu değerleri komşu aralıklara yerleştirme işlemi sonucu ayrıklaştırma sürecini de gerçekleştirdiğini ifade etmiştir.

Holte (1989) daha önceki çalışmasında, sürekli değerleri sonlu sayıdaki bir aralığa dönüştürme işlemi sırasında meydana gelebilecek aynı sınıfa ait örneklerin "saf/katışıksız" birçok aralık gerektirmesi şeklindeki soruna çözüm üretmek zorunda kalmış ve tüm aralıkların aynı sınıfa ait en az 6 örnek içermesi koşulunun gerekliliğini belirtmiştir.

1RD algoritması her bir aralığa yukarıda da belirtilen minimum sayıda değer gelecek şekilde aralıkların belirlenmesiyle başlar. Bu aralıklara nitelik değerlerinin eklenmesi sonucu işlemlere başta elde edilen kesim noktalarının kaydırılması şeklinde devam edilir (Su ve Hsu, 2005).

Algoritma şu şekilde ifade edilebilir:

a. Veriler sıralanır.
b. Sürekli değerlerin aralığı, aynı sınıf etiketine sahip en az 6 nitelik değeri içerecek şekilde sonlu sayıdaki ayrık aralıklara bölünür. Bu aralıkların sınırları niteliklere ait değerlerin ilişkili oldukları sınıf etiketlerine göre belirlenir.

c. Sınırda yer alan değer bir önceki komşu aralığın çoğunluk sınıfından farklı bir sınıfına ait olana kadar, sınır kaydırılarak sınır belirlenmesine devam edilir.
d. Sonuç olarak elde edilen komşu aralıkların çoğunluk sınıfları aynı ise bu aralıkların birleştirilme işlemi gerçekleştirilir ve ayrıklaştırma işlemine son verilmiş olur.

1RD öntemi ile ilgili bir örnek Tablo 3.3'te verilmektedir (Ismail, 2003).

Tablo 3.3: 1RD yöntemi sonucu elde edilen aralıklar (Ismail, 2003)

Aralık	Aralık 1										Aralık 2			
Örnek	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Değer	64	65	68	69	70	71	72	72	75	75	80	81	83	85
Sınıf	S	L	S	S	S	L	L	S	S	S	L	S	S	L

Yukarıda verilen tablonun elde edilebilmesi için aşağıdaki algoritma takip edilir:

- Öncelikle tüm örnek değerler artan düzende sıralanır.
- Her bir aralık aynı sınıf etiketine ait 6 örnek içerecek şekilde ayrık aralıklar oluşturulur (Bunun sonucu olarak ilk sınır Ssınıf etiketinden altı nitelik değeri içerecek şekilde 9. örneğin sonuna gelmiştir).
- Ancak komşu aralıkların sınır değerleri incelendiğinde 2. aralıkta yer alan ilk örnek değeri bir önceki aralığın çoğunluk sınıf etiketi ile aynı olursa sınır kaydırılır (Buradaki örnekte 9 ve 10. değerler komşu aralıkların sınır değerleridir. Her iki sınır değeri de S sınıfından olduğundan sınır 10. değerden sonraya çekilmiştir. Tekrar control edildiğinde 2. aralığın ilk değeri yani 11. örnek L sınıfına ait olduğundan ayrıklaştırma işlemi son bulmuştur).
- Aynı kontroller yeni sınırlar için de yapılarak sınır değerlerin bir önceki komşu sınıfların çoğul sınıf etiketinden farklı olması sağlanır.
- Eğer en son durumda iki komşu aralık çifti aynı çoğul sınıf etiketine sahip ise bu iki aralık birleştirilir.

Sonuç olarak 77.5'e eşit ve küçük değerler aralık 1'e bunun sonucu olarak da S sınıfına, 77.5'ten büyük değerler de aralık 2'ye yani L sınıfına ait olacak şekilde iki kategoriye ayrılmıştır.

3.3.7. Sınıf-Nitelik Bağımlı Ayrıklaştırma (CADD)

Sınıf-Nitelik Bağımlı Ayrıklaştırma (Class-Attribute Dependent Discretizer:CADD) algoritması 1995 yılında John Y. Ching ve çalışma arkadaşları tarafından geliştirilmiştir. CADD yöntemi de sınıf bilgisi kullanımı nedeniyle denetimli ayrıklaştırma yöntemleri kategorisinde yer almaktadır. Bu yöntemde en iyi ayrıklaştırma sonucunun elde edilebilmesi için sınıf-nitelik bağımlılık bilgisi bir kriter olarak kullanılmaktadır (Ching ve diğ., 1995).

Yi ve diğ. (2011) çalışmalarında sınıf-nitelik bağımlılık bilgisini kullanan yöntemlerde; dolayısı ile bu yöntemde de, hedef, sınıf ve nitelik değerlerinin iki değişken gibi alınarak, bunlar arasındaki bağımlılığı ölçmek ve buna bağlı olarak sınır noktaları arasındaki ilişkiyi tanımlamak için bazı değerlerin kriter olarak kullanıldığını belirtmişlerdir. Bahsi geçen bu hedef sınıf ve nitelik arasındaki bağımlılık ölçüsünü belirlemek için Tablo 3.4'te belirtilen sınıf-nitelik olasılık tablosu (2D Quanta Matrix) oluşturularak CAIR (Class-Attribute Interdependence Redundancy) adı verilen değer hesaplanmaktadır.

Tablo 3.4: 2D Quanta Matrix Tablosu (Ching ve diğ., 1995)

Sınıf	$[e_0, e_1]$ ...	$(e_{r-1}, e_r]$ ...	$(e_{n-1}, e_n]$	Toplam
c_1	q_{11}	q_{1r}	q_{1n}	M_{1+}
.	.	.	.	.
.	.	.	.	.
c_i	q_{i1}	q_{ir}	q_{in}	M_{i+}
.	.	.	.	.
.	.	.	.	.
c_S	q_{S1}	q_{Sr}	q_{Sn}	M_{S+}
Toplam	M_{+1}	M_{+r}	M_{+n}	M

q_{ir}, i. sınıfa ait $(e_{r-1}, e_r]$ aralığında bulunan sürekli niteliklerin sayısı

M_{i+}, i. sınıfa ait tüm nesnelerin sayısı

M_{+r}, $(e_{r-1}, e_r]$ aralığında yer alan A niteliğine ait sürekli değerlerin toplam sayısı

Chaoqun ve diğ. (2011), CAIR değerini hesaplamak için kullanılacak A niteliğine ait değerlerin aynı anda $(e_{r-1}, e_r]$ aralığında bulunma ve c_S sınıfına ait olma durumlarının gerçekleşme olasılığını şu şekilde belirlemişlerdir:

$$p_{ir} = q_{ir} / M \tag{3.9}$$

Bunun yanı sıra A niteliği değerlerinin c_i sınıfına ait olma olasılığı,

$$p_{i+} = M_{i+} / M \tag{3.10}$$

biçiminde hesaplanır. A niteliği değerlerinin $(e_{r-1}, e_r]$ aralığına ait olma olasılığı ise,

$$p_{+r} = M_{+r} / M \tag{3.11}$$

şeklinde formülize edilir.

CAIR (Class-Attribute Interdependence Redundancy) değeri, sınıf-nitelik arası bağımlılık değerini belirleyen ve Tablo 3.4'te verilen Quanta matrix tablosundan yararlanılarak hesaplanan bir değerdir. CAIR değeri şu şekilde formülize edilmektedir:

$$CAIR = \frac{\sum_{i=1}^{s}\sum_{r=1}^{n} p_{ir} \log_2 \frac{p_{ir}}{p_{i+}p_{+r}}}{\sum_{i=1}^{s}\sum_{r=1}^{n} p_{ir} \log_2 \frac{1}{p_{ir}}} \tag{3.12}$$

CAIR formülünün tanımına göre nitelik sayıları, sınıf ve nitelik arasındaki bağımlılık ilişkisi yok edilmeden azaltılabilmekte ve böylece ayrıklaştırma işlemi, karşılaşılan çok sayıdaki gereksiz nitelik değerinin kaldırılabildiği bir süreç haline dönüşebilmektedir. Bu da CAIR değerinin bir kriter olarak alınmasını sağlamaktadır (Li ve diğ., 2011).

Ching ve diğ. (1995), CADD algoritmasını özetle şu şekilde ifade etmişlerdir:

a. Bir eğitim veri kümesi içerisinden alınan niteliklere ait tekil değerler artan sırada dizilir.
b. Başlangıç için bir aralık sayısı belirlenir.
c. Belirlenen aralık sayısına göre başlangıç aralıkları oluşturulur, sınırlar belirlenir ve Tablo 3.4'te verilen Quanta matrix tablosu oluşturulur.

d. Oluşturulan bu tabloya göre CAIR değeri hesaplanır.
e. Sınır değerler üzerinde oynamalar yapılarak Quanta matrix tablosunun düzeni değiştirilir. Buna göre yeni CAIR değerleri hesaplanır.
f. Sınır değerlerdeki düzenlemeler ve buna bağlı bağımlılık kriterlerinin belirlenmesi sonucu maksimum kazanç sağlayan düzenleme seçilerek sınır değerler bu düzenlemeye göre yeniden belirlenir. Buna bağlı olarak da yeni Quanta matrix tablosu elde edilir.
g. Tüm bu işlemler bağımlılık kriteri üzerinde sağlanan kazancın durmasına kadar tekrar ettirilir.
h. Son olarak elde edilen tüm aralık çiftleri istatistiksel teste tabi tutulur. Eğer aralık çiftlerinin birleştirilmesi bağımlılık kriterini etkilemiyor ise bu aralıklar birleştirilir.

3.3.8. Sınıf-Nitelik Bağımlılığı Maksimizasyounu (CAIM)

Sınıf nitelik bağımlılığına bağlı olarak sonuç üreten bir diğer algoritma Lukasz Kurgan ve Krzysztof Cios tarafından geliştirilen Sınıf-Nitelik Bağımlılığı Maksimizasyonu (Class-Attribute Intedependence Maximization:CAIM) algoritmasıdır. Bu algoritma ayrıklaştırma sonrası elde edilecek ayrık aralık sayısını otomatik olarak kendisi belirleyerek kullanıcıdan herhangi bir değer girmesini beklemez. Dolayısıyla kullanıcı etkileşimi azaltılmış olur (Kurgan ve Cios, 2003).

Kurgan ve Cios (2004), CAIM algoritmasının amaçlarını şu şekilde sıralamışlardır:

a. Sınıf etiketi ve sürekli değerli nitelik arasındaki bağımsızlığı en üst düzeye çıkarmak,
b. Mümkün olan en küçük aralık sayısını elde etmek,
c. Makul bir maliyet ile ayrıklaştırma işlemini gerçekleştirmek.

CAIM algoritması sınıf-nitelik bağımlılığını hesaplayabilmek için CADD algoritmasında olduğu gibi Tablo 3.4'ü kullanır. Buna göre CAIM kriteri oluşturulur. Bu kriter şöyle formüle edilir (Tsai ve diğ., 2008):

$$CAIM = \frac{\sum_{r=1}^{n} \frac{max_r^2}{M_{+r}}}{n} \tag{3.11}$$

max_r, r. sütunda bulunan en büyük değer

n, aralık sayısı

M_{+r}, $(e_{r-1}, e_r]$ aralığında yer alan A niteliğine ait sürekli değerlerin toplam sayısı

M örnek sayısı, S sınıflar ve A_i'ler nitelik değerleri olmak üzere CAIM algoritması şu şekilde ifade edilmektedir (Kurgan ve Cios, 2004):

a. A_i nitelik değerleri içerisindeki en küçük ve en büyük değerler belirlenir. Bu değerler Tablo 3.4'te bahsi geçen aralıkların en büyük değeri e_0 ve en küçük değeri e_n olacaktır.
b. Nitelik değerleri artan düzende sıralanır ve mümkün olan tüm aralık sınırları belirlenir.
c. Belirlenen bu aralıklar ile taslak bir aralıklar kümesi hazırlanarak GenelCAIM değeri başlangıç olarak sıfıra eşitlenir.
d. Başlangıç değeri olarak k=1 belirlenir.
e. Belirlenen taslak aralıklar içerisinde yer almayan yeni geçici sınırlar eklenir.
f. Tüm bu geçici eklemelerin yapılarak eklenen her aralık sonrası Tablo 3.4'ün yardımı ile CAIM değerleri hesaplanır.
g. Hesaplanan CAIM değerleri arasından en yüksek olanı seçilir.
h. Başta sıfır olarak belirlenen GenelCAIM değeri yeni bulunan değer ile değiştirilir. Bundan sonra elde edilen CAIM değeri en son atanan GenelCAIM değeri ile karşılaştırılarak işlem devam eder. Eğer bu adımda bulunan maksimum CAIM değeri bir önce atanan GenelCAIM değerinden küçük ise işleme son verilir. Aksi halde bir sonraki adıma geçilir.
i. En yüksek CAIM değerinin elde edildiği aralık belirlenen aralıklar dizisinin içine katılır. k değeri bir arttırılarak e. adıma gidilir ve sonraki işlemler devam ettirilir.

Tüm olası CAIM kriter değerleri taramasının tamamlanmasından sonra ayrıklaştırma işleminin sonucu olarak ayrık aralıklar elde edilmiş olur.

4. MALZEME VE YÖNTEM

4.1. PROBLEMİN BELİRLENMESİ

Veri madenciliği çalışmalarında, veri madenciliği yöntemlerinin veri kümesi üzerine uygulanması ve sonuçların yorumlanması kadar üzerinde önemle durulması gereken bir başka süreç de veri ön işleme adımıdır. Çünkü bilinmektedir ki düzgün veri nitelikli bilgi ve doğru karara götürür.

Tez çalışmasına başlamadan önce veri madenciliği çalışmaları incelenmiş, çalışılacak konunun ne olabileceği konusunda araştırma yapılmıştır. Bu alanda yapılan çalışmalar incelendiğinde, ağırlıklı olarak veri madenciliği uygulamalarına yönelik olduğu belirlenmiştir. Ardından veri madenciliği sürecine ait adımlar incelenmiş ve veri önişleme işlemlerinin öneminin vurgulanmasında eksik kalındığı görülmüştür. Tez çalışması kapsamında hem bu adımlar hakkında bir çalışma yaparak elde edilecek sonuçlar ile literatüre katkıda bulunmak; hem de bu adımların önemine dikkat çekmek istenmiştir.

Bu hedefle yola çıkılmış olup literatürde yapılan incelemeler sonucu, ön işleme adımları arasında yer alan veri ayrıklaştırma yöntemleri üzerine odaklanılmıştır. Birçok uluslararası makalede, ayrıklaştırma yöntemleri arasındaki farklılıkları ortaya koyan çalışmalar, araştırmacıların geliştirdikleri yöntemin diğer birkaç yöntem ile karşılaştırılması ile sınırlı kaldığı görülmüştür. Ülkemizde ise böyle bir çalışmaya rastlanmamıştır.

Buradan hareket ile “Belirlenen sekiz farklı ayrıklaştırma yönteminin uygulanması ile elde edilen aralık sayıları ve kategorik değişkenlerin karşılaştırılması sonucunda yöntemlerin tutarlı çalışma oranları, birbirlerine göre avantaj ve dezavantajları nelerdir?” sorusu tez çalışmasının ana problemi olarak belirlenmiştir.

4.2. KNOWLEDGE EXTRACTION BASED ON EVOLUTIONARY LEARNING (KEEL) YAZILIMI

Çalışma kapsamında, karşılaştırması yapılacak veri ayrıklaştırma yöntemlerinin veri kümeleri üzerinde uygulanabilmesi için Knowledge Extraction based on Evolutionary Learning (KEEL) (Alcalá-Fdez ve diğ., 2009) yazılımı kullanılmıştır.

KEEL yazılımı java platformunda geliştirilmiş; regresyon, kümeleme, sınıflandırma gibi farklı veri madenciliği uygulamalarına olanak sağlayan ve bu yöntemler ile değerlendirmeler yapılabilmesini destekleyen açık kaynak kodlu bir yazılımdır. İspanyol Ulusal Projeleri kapsamında TIC2002-04036-C05, TIN2005-08386-C05 and TIN2008-06681-C06 numaraları ile SCI^{2}S, Ayma, GRSI, Intelligent Systems and Data Mining, Metrology and Models araştırma gruplarının ortaklaşa geliştirmekte olduğu bir projedir. Bu yazılım oluşturulurken hedef kullanıcı kitlesi olarak özellikle araştırmacı ve öğrenci grupları belirlenmiştir.

Yazılım bünyesinde; Veri Yönetimi, Deney Tasarımı, Dengesiz Deney Tasarımı, İstatistiksel Testler ve Eğitimsel Deneyler olmak üzere beş farklı fonksiyon bulundurmaktadır. Tez çalışması için kullanılan 'Veri Yönetimi' fonksiyonu kapsamında farklı formatlardaki veri kümelerinin yazılımın formatı olan .dat uzantılı veri kümesi dosyalarına veya tam tersi yönde format değişikliği yapılabilmekte, yazılıma dahil olarak gelen veri kümelerinin yazılıma entegre edilmesi veya içerdeki bir veri kümesinin dışa aktarımı gerçekleştirilebilmektedir. Yine çalışma kapsamında kullanılan 'Deney Tasarımı' fonksiyonu ile deneye eklenecek veri kümelerine veri madenciliği yöntemlerinin uygulanması sağlanmaktadır. Şekil 4.1 ve Şekil 4.2'de Keel yazılımının arayüz görüntüleri yer almaktadır.

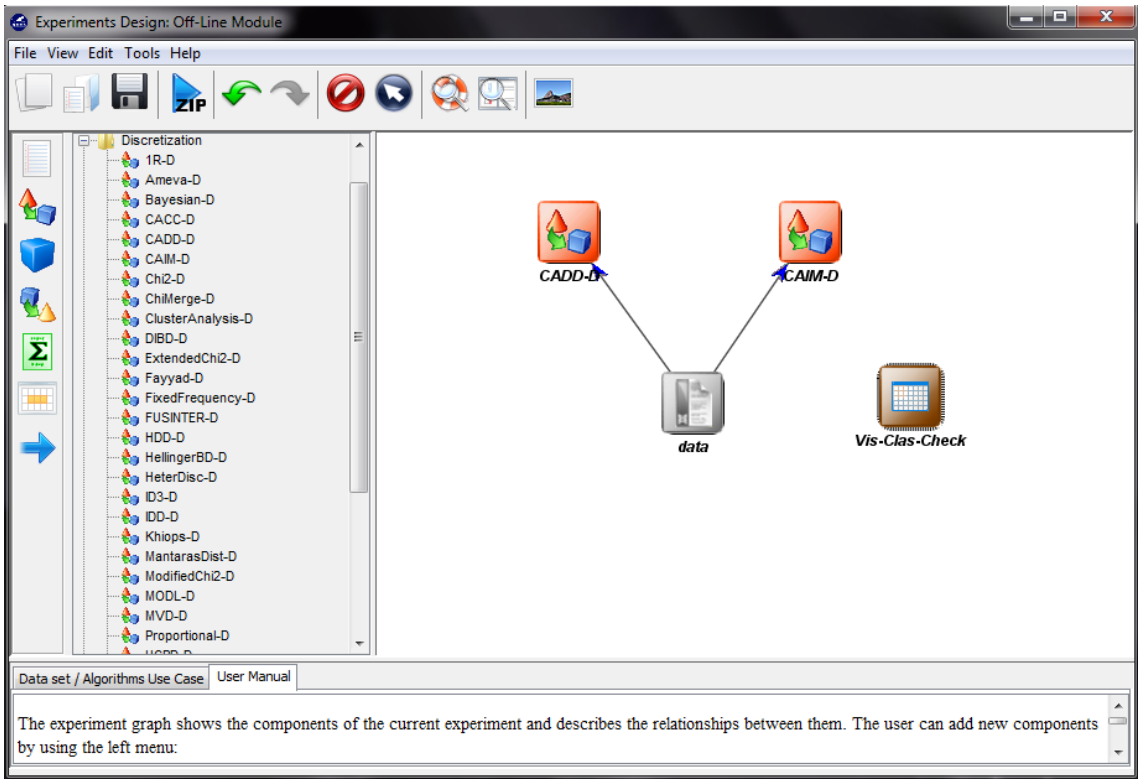
Veri önişleme adımlarından veri ayrıklaştırma yöntemi olarak 30 farklı yöntem uygulanabilmekte olup bu yöntemler şunlardır:

1. Uniform Width Discretizer
2. Uniform Frequency Discretizer
3. Fayyad Discretizer
4. Iterative Dicotomizer 3 Discretizer

5. Bayesian Discretizer
6. Mantaras Distance-Based Discretizer
7. Unparametrized Supervised Discretizer
8. Chi-Merge Discretizer
9. Chi2 Discretizer
10. Ameva Discretizer
11. Zeta Discretizer
12. Class-Atribute Dependent Discretizer
13. Class-Atribute Interdependence Maximization
14. Extended Chi2 Discretizer
15. Fixed Frequency Discretizer
16. Khiops Discretizer
17. Modified Chi2 Discretizer
18. MODL Discretizer
19. 1R Discretizer
20. Proportional Discretizer
21. Discretization Algorithm Based on a Heterogeneity Criterion
22. Hellinger-based Discretizer
23. Distribution-Index-Based Discretizer
24. Unsupervised Correlation Preserving Discretization
25. Interval Distance-Based Method for Discretization
26. Discretization algorithm based on Class-Attribute Contingency Coefficient
27. Hypercube Division-Based
28. Cluster Analysis
29. Multivariate Discretization
30. FUSINTER

Yazılım uygulanacak ayrıklaştırma yöntemleri için hazırlanan deney setini sıkıştırılmış klasör olarak kaydetmekte klasör içerisinde oluşturulan scripts klasörünün içerisinde yer alan RunKeel.jar dosyasının çalıştırılması ile oluşturulan bu dosyaya elde edilen sonuçlar text dosyası olarak yüklenmektedir.

Şekil.4.1: Keel yazılım aracının kullanıcı arayüz görüntüsü



Şekil.4.2: Keel yazılım aracında örnek bir deney tasarımı

4.3. VERİ TOPLAMA

Çalışmada kullanılacak veri kümesi için KDnuggets, Amazon Web Services Public Data Sets, Datasource Handbook, DataMarket, Carnegie Mellon University StatLib Datasets gibi bazı çalışmalarda kullanılmış veya bazı kurumlar tarafından kullanıma sunulmuş ücretsiz veri kümelerinin bulunabileceği web sayfaları araştırılmış, bu araştırma sonucu bulunan bazı veri kümeleri kullanılacak KEEL yazılımına entegre edilmeye çalışılmış ancak başarılı olunamamıştır. Bunun sonucu olarak ayrıklaştırma yöntemlerinin uygulanabilmesi için KEEL yazılımı içerisinde hazır olarak gelen Wisconsin veri kümesi kullanılmıştır (Alcalá-Fdez ve diğ., 2011). Bu veri kümesinin seçilmesinin nedeni nitelik değerlerinin sürekli sayısal değerlerden meydana gelmesidir.

4.4. VERİ KÜMESİ

Çalışma kapsamında kullanılan veri kümesi içerisindeki veriler, Wisconsin Hastaneler Üniversitesi'nde meme kanseri teşhisi konulup ameliyat geçirmiş hastalardan elde edilmiş olup, bazı nitelik değerlerine göre hastalardan alınan tümör örneklerinin iyi veya kötü huylu olup olmadıklarını tespit etmek amaçlanmıştır.

Tablo 4.1'de de belirtildiği üzere veri kümesi içerisinde 9 nitelik ve 2 sınıf bulunmakla beraber her bir nitelik değeri tablo 4.2'de görülebileceği gibi 1 ile 10 arasında tamsayı değeri almaktadır.

Aşağıdaki tablolarda veri kümesine ait temel özellikler, içerisinde yer alan verilere ait nitelikler ve değer aralıkları belirtilmiştir.

Tablo.4.1: Veri kümesi genel özellikleri

Nitelik Sayısı	: 9	Gerçek/Tamsayı/Kategorik	: 0/9/0
Sınıf Sayısı	: 2	Kayıp değer var mı?	: Evet
Toplam Örnek Sayısı	: 699	Kayıp değerler dışındakli örnek sayısı	: 683

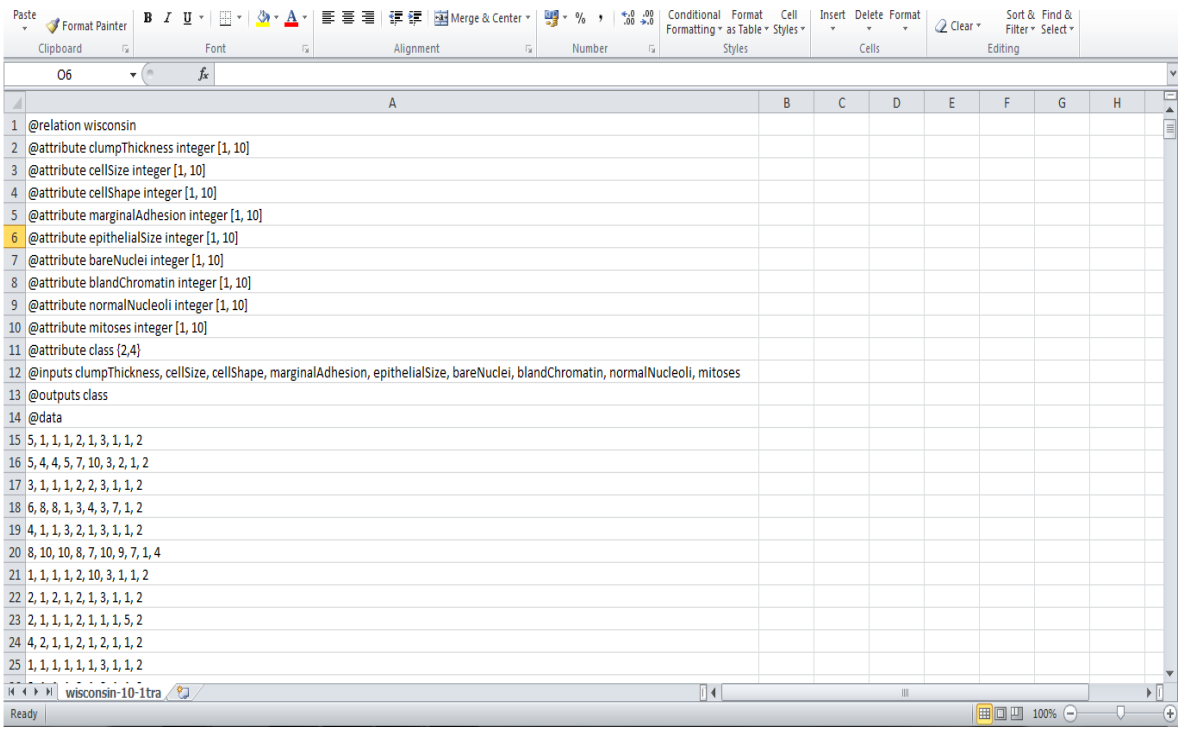
Tablo.4.2: Veri kümesine ait nitelikler ve etki alanları

Nitelik	**Etki Alanı**
ClumpThickness (Parça Kalınlığı)	[1, 10]
CellSize (Hücre Boyutu)	[1, 10]
CellShape (Hücre Şekli)	[1, 10]
MarginalAdhesion (Marjinal Adezyon)	[1, 10]
EpithelialSize (Epitel Boyutu)	[1, 10]
BareNuclei (Açık Çekirdek)	[1, 10]
BlandChromatin (Uysal Kromatin)	[1, 10]
NormalNucleoli (Normal Çekirdekçik)	[1, 10]
Mitoses (Mitoz)	[1, 10]
Sınıf	{2,4}

Veri kümesinin yazılımda kullanılabilirliği için veri kümesi dosyasının keel formatında olması gerekmektedir. Keel formatında dosyalar .dat uzantılı olup her veri dosyası aşağıdaki düzene sahiptir.

@relation: Veri kümesinin adı
@attribute: Her bir niteliğin açıklaması
@inputs: Giren niteliklerin isimleri ile listesi
@output: Çıkan nitelik isimleri
@data: Veri başlangıç etiketi

Bu çalışma kapsamında veri kümesinin formatında herhangi bir değişiklik yapılmamış olup, çıktılar yazılım sayesinde istenilen dosya formatına da dönüştürülmüştür. Şekil 4.3'te kullanılan veri kümesine ilişkin örnek görüntü yer almaktadır.



	A
1	@relation wisconsin
2	@attribute clumpThickness integer [1, 10]
3	@attribute cellSize integer [1, 10]
4	@attribute cellShape integer [1, 10]
5	@attribute marginalAdhesion integer [1, 10]
6	@attribute epithelialSize integer [1, 10]
7	@attribute bareNuclei integer [1, 10]
8	@attribute blandChromatin integer [1, 10]
9	@attribute normalNucleoli integer [1, 10]
10	@attribute mitoses integer [1, 10]
11	@attribute class {2,4}
12	@inputs clumpThickness, cellSize, cellShape, marginalAdhesion, epithelialSize, bareNuclei, blandChromatin, normalNucleoli, mitoses
13	@outputs class
14	@data
15	5, 1, 1, 1, 2, 1, 3, 1, 1, 2
16	5, 4, 4, 5, 7, 10, 3, 2, 1, 2
17	3, 1, 1, 1, 2, 2, 3, 1, 1, 2
18	6, 8, 8, 1, 3, 4, 3, 7, 1, 2
19	4, 1, 1, 3, 2, 1, 3, 1, 1, 2
20	8, 10, 10, 8, 7, 10, 9, 7, 1, 4
21	1, 1, 1, 1, 2, 10, 3, 1, 1, 2
22	2, 1, 2, 1, 2, 1, 3, 1, 1, 2
23	2, 1, 1, 1, 2, 1, 1, 1, 5, 2
24	4, 2, 1, 1, 2, 1, 2, 1, 1, 2
25	1, 1, 1, 1, 1, 1, 3, 1, 1, 2

Şekil 4 3: Wisconsin veri kümesinin ilk 11 satırının MS Excel2010 çalışma sayfasındaki örnek görüntüsü

4.5. YÖNTEM

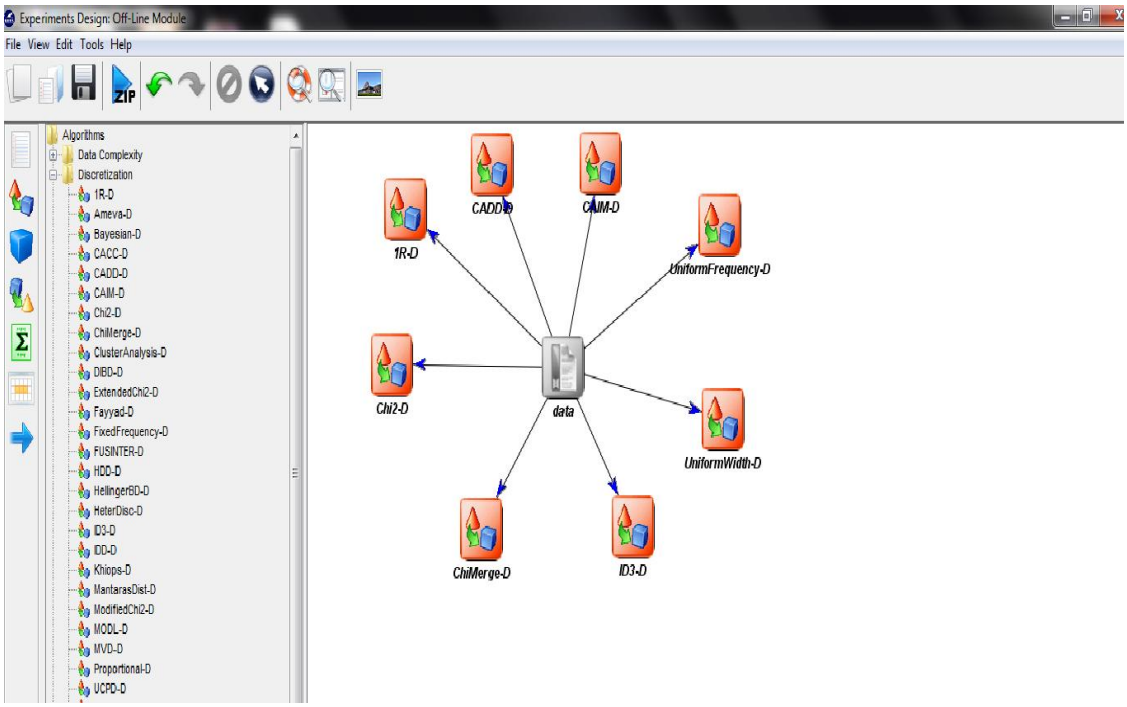
Çalışma; literatürün taranması, veri kümesinin seçimi, uygulanacak veri ayrıklaştırma yöntemlerinin belirlenmesi, ayrıklaştırma yöntemlerinin veri kümesi üzerine uygulanması ve sonuçların yorumlanması şeklinde beş adımdan oluşmaktadır.

Literatür taraması için İ.Ü Merkez Kütüphanesi ve Compiegne Teknik Üniversitesi (Fransa) Kütüphanesi'nden yararlanılmış; EbscoHost, ScienceDirect, Springer Link, ULAKBİM, Proquest elektronik veri tabanları, IEEE Xplore dijital kütüphanesi, Google Akademik, Google Books arama motorları kullanılmıştır. Elektronik taramalarda data mining, data mining methods, data cleaning, data cleansing, data discretization, data discretization tools, veri ayrıklaştırma vb. kelimeler anahtar kelime olarak kullanılmıştır.

Veri kümesinin seçimi ile ilgili olarak izlenen yol veri toplama başlığı altında ayrıntılı olarak ele alınmıştır.

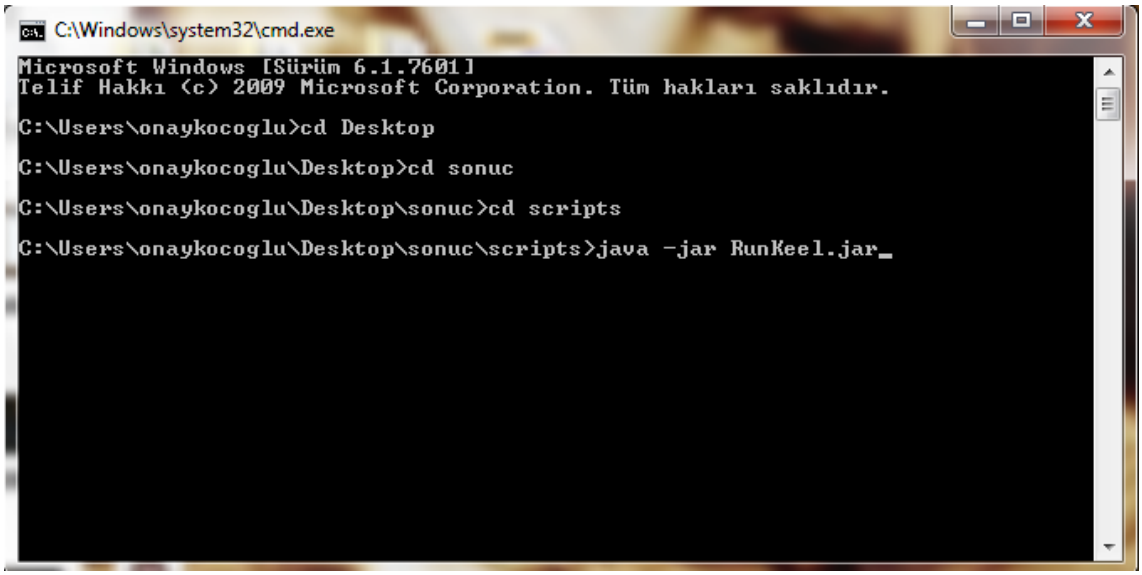
Uygulanacak veri ayrıklaştırma yöntemlerinin seçimi için özellikle sık kullanılan ayrıklaştırma yöntemlerinin tercih edilmesine, belirtilen ayrıklaştırma yöntemleri kategorilerine, ele alınacak yöntem sayısı da göz önünde tutularak, örnek olacak şekilde seçilmesine özen gösterilmiştir.

Veri ayrıklaştırma yöntemlerinin uygulanması aşamasında öncelikle WEKA, YALE, KEEL olmak üzere bazı yazılım araçları incelenmiştir. WEKA ve YALE yazılımlarında izin verilen yöntem sayıları sınırlı sayıda olduğundan KEEL yazılım aracı tercih edilmiştir. Bu yazılım aracında yer alan deney tasarımı fonksiyonu kullanılarak, veri kümesi ve uygulanacak yöntemler tasarım alanına taşınmış, veri kümesi ve yöntemler arasında bağlantılar kurulmuştur. Şekil 4.4'te elde edilen tasarım yer almaktadır.

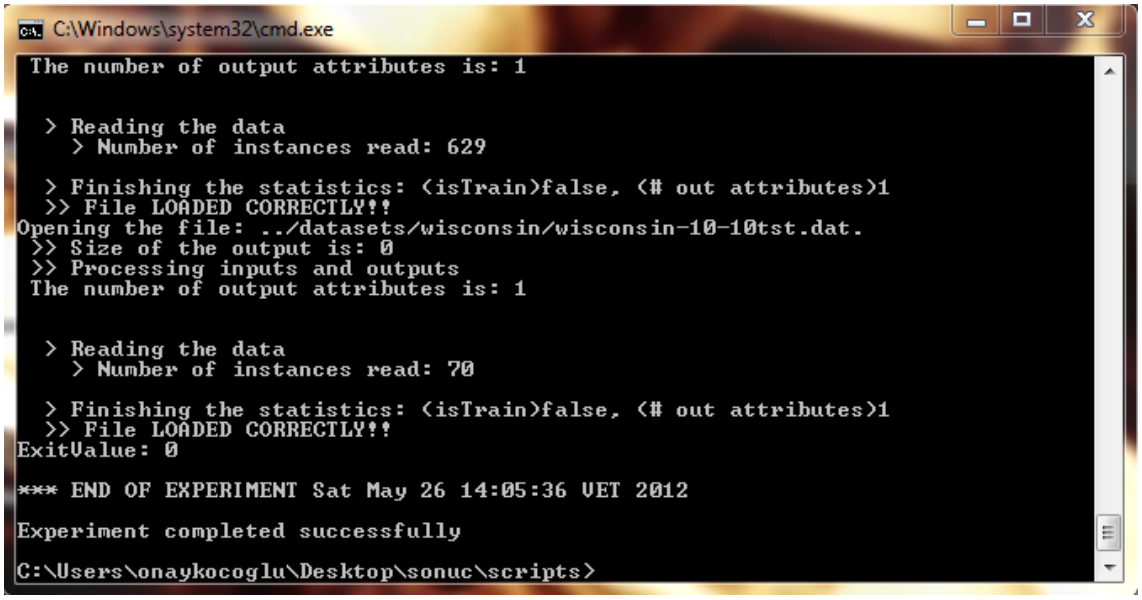


Şekil 4.4: 'Deney Tasarımı' fonksiyonu kullanılarak hazırlanan uygulama

Elde edilen tasarım sıkıştırılmış klasör olarak kaydedilmiştir. Uygulamanın sonuçlarını elde edebilmek için ana klasörün altında yer alan scripts dosyası içerisindeki KeelRun.jar uzantılı dosya çalıştırılmıştır. Bu dosyanın çalıştırılmasına ilişkin görüntüler Şekil 4.5 ve Şekil 4.6'da verilmiştir.



Şekil 4.5: Tasarlanan uygulamanın sonuçlarının yüklenmesi için RunKeel.jar dosyasının çalıştırılması



Şekil 4. 6: Tasarlanan uygulamanın başarı ile yüklendiği bilgisinin alındığı ekran görüntüsü

Tasarımın çalıştırılması sonrası elde edilen sonuçlar results klasörünün altında toplanmaktadır. Elde edilen sonuçların karşılaştırılmasının görsel olarak daha rahat yapılabilmesi için her bir yönteme ait .dat uzantılı dosya içerisinde yer alan sonuçlar MS Excel2010 çalışma sayfasında yan yana sütunlara taşınmıştır.

Bu çalışma kapsamında ayrıklaştırma yöntemlerinin uygulanması sonucu elde edilen kesim noktaları sayıları ve kategorik değişkenlerin birbirleri ile olan benzerliklerinin oranları karşılaştırılarak değerlendirme yapılmıştır.

5. BULGULAR

Bulgular, Wisconsin veri kümesine her bir yöntemin uygulanması sonucu elde edilen kesim noktaları sayısı ve belirtilen yöntemlerin uygulanması sonrası veri kümesinde elde edilen kategorik değişkenler ve diğer bulgular olmak üzere üç farklı başlık altında belirtilmiştir.

5.1. KESİM NOKTASI SAYILARINA AİT BULGULAR

Tablo 5.1: 1RD ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		Kesim Noktası Sayısı	Kesim Noktası Değerleri
ClumpThickness	:	0	-
CellSize	:	0	-
CellShape	:	0	-
MarginalAdhesion	:	0	-
EpithelialSize	:	0	-
BareNuclei	:	1	1.0
BlandChromatin	:	0	-
NormalNucleoli	:	0	-
Mitoses	:	0	-

Tablo 5.1'de veri kümesi üzerinde 1RD algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. Buna göre BareNuclei niteliğine ait bir kesim noktası ve buna bağlı olarak elde edilen aralıktaki değerler 1.0 olarak belirlenen değerden küçük ve büyük olmak üzere iki kategori altında toplanmıştır. Diğer nitelikler için elde edilen sonuçlara bakıldığında ise kesim noktası elde edilmemiş buna bağlı olarak tek bir kategorik değişken oluşturulmuştur.

Tablo 5.2: CADD ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		Kesim Noktası Sayısı	Kesim Noktası Değerleri
ClumpThickness	:	1	1.0
CellSize	:	1	1.0
CellShape	:	1	1.0
MarginalAdhesion	:	1	1.0
EpithelialSize	:	1	1.0
BareNuclei	:	1	1.0
BlandChromatin	:	1	1.0
NormalNucleoli	:	1	1.0
Mitoses	:	0	-

Tablo 5.2'de veri kümesi üzerinde CADD algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. Buna göre Mitoses niteliğine ait kesim noktası elde edilmemiş buna bağlı olarak bu nitelik için tek bir kategorik değişken oluşturulmuştur. Diğer nitelikler için sonuçlar incelendiğinde ise bir kesim noktası oluşturulduğu belirlenmişitr. Buna göre niteliklerin değerlerinin ait oldukları aralıklar 2'ye bölünmüş ve değerler, tabloda belirtilen kesim noktası değerinden küçük ve büyük olmak üzere iki kategori altında toplanmıştır. Ayrıca kesim noktalarına ait değerlerin de tüm nitelikler için aynı olduğu görülmektedir.

Tablo 5.3: CAIM ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		Kesim Noktası Sayısı	Kesim Noktası Değerleri
ClumpThickness	:	1	5.5
CellSize	:	1	3.5
CellShape	:	1	3.5
MarginalAdhesion	:	1	3.5
EpithelialSize	:	1	2.5
BareNuclei	:	1	2.5
BlandChromatin	:	1	3.5
NormalNucleoli	:	1	2.5
Mitoses	:	1	1.5

Tablo 5.3'te veri kümesi üzerinde CAIM algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. Buna göre tüm nitelikler için sonuçlar incelendiğinde bir kesim noktası oluşturulmuştur. Buna göre niteliklerin aldığı değerlerin ait oldukları aralıklar 2'ye bölünmüş ve değerler, tabloda belirtilen kesim noktası değerlerinden küçük ve büyük olmak üzere iki kategori altında toplanmıştır.

Tablo 5.4: Chi2 ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		**Kesim Noktası Sayısı**	**Kesim Noktası Değerleri**
ClumpThickness	:	0	-
CellSize	:	1	3.5
CellShape	:	1	1.5
MarginalAdhesion	:	0	-
EpithelialSize	:	1	2.5
BareNuclei	:	1	1..5
BlandChromatin	:	1	3.5
NormalNucleoli	:	1	2.5
Mitoses	:	0	-

Tablo 5.4'te veri kümesi üzerinde Chi2 algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. Buna göre ClumpThickness, MarginalAdhesion ve Mitoses niteliklerine ait kesim noktası elde edilmemiş, buna bağlı olarak bu nitelikler için tek bir kategorik değişken oluşturulmuştur. Diğer nitelikler için sonuçlar incelendiğinde ise her bir nitelik için tek bir kesim noktası oluşturulduğu belirlenmişitr. Buna göre niteliklerin değerlerinin ait oldukları aralıklar 2'ye bölünmüş ve değerler, kesim noktası değerinden küçük ve büyük olmak üzere iki kategori altında toplanmıştır.

Tablo 5.5: ChiMerge ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		Kesim Noktası Sayısı	Kesim Noktası Değerleri
ClumpThickness	:	1	6.5
CellSize	:	1	3.5
CellShape	:	1	1.5
MarginalAdhesion	:	1	1.5
EpithelialSize	:	1	2.5
BareNuclei	:	1	1.5
BlandChromatin	:	1	3.5
NormalNucleoli	:	1	2.5
Mitoses	:	1	1.5

Tablo 5.5'te veri kümesi üzerinde ChiMerge algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. Buna göre tüm nitelikler için sonuçlar incelendiğinde her bir nitelik için bir kesim noktası oluşturulmuştur. Buna göre niteliklerin aldığı değerlerin ait oldukları aralıklar 2'ye bölünmüş ve değerler, kesim noktası değerinden küçük ve büyük olmak üzere iki kategori altında toplanmıştır.

Tablo 5.6: ID3 ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		Kesim Noktası Sayısı	Kesim Noktası Değerleri
ClumpThickness	:	8	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5
CellSize	:	9	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5
CellShape	:	8	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5
MarginalAdhesion	:	8	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 8.5, 9.5
EpithelialSize	:	9	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5
BareNuclei	:	8	1.5, 2.5, 3.5, 4.5, 5.5, 7.5, 8.5, 9.5
BlandChromatin	:	7	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5
NormalNucleoli	:	9	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5
Mitoses	:	8	1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 9.0

Tablo 5.6'te veri kümesi üzerinde ID3 algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. Tablodan da görüldüğü üzere

nitelikler için 7 ile 9 arasında kesim noktası yani 8 ile 10 arasında kategorik değişken elde edilmiştir. Elde edilen bu kategorik değişkenler tabloda belirtilen değer aralıklarına göre atanmıştır.

Tablo 5.7: Eşit Frekanslı ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		**Kesim Noktası Sayısı**	**Kesim Noktası Değerleri**
ClumpThickness	:	7	1.5, 2.0, 3.5, 4.5, 5.5, 6.0, 9.5
CellSize	:	5	1.5, 2.5, 3.0, 5.0, 9.5
CellShape	:	5	1.5, 2.5, 3.0, 6.5, 8.5
MarginalAdhesion	:	5	1.5, 2.5, 3.5, 4.0, 8.5
EpithelialSize	:	5	1.5, 2.5, 3.5, 4.0, 6.5
BareNuclei	:	4	1.5, 2.5, 4.5, 9.5
BlandChromatin	:	6	1.5, 2.5, 3.5, 4.5, 5.0, 7.5
NormalNucleoli	:	4	1.5, 2.0, 5.5, 9.5
Mitoses	:	2	1.5, 2.5

Tablo 5.7'de veri kümesi üzerinde eşit frekanslı ayrıklaştırma algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri görülmektedir. ID3 algoritmasında olduğu gibi her bir nitelik için birden fazla kesim noktası elde edilmiştir. Tablodan da görüldüğü üzere nitelikler için 2 ile 7 arasında kesim noktası yani 3 ile 8 arasında kategorik değişken elde edilmiştir. Tabloda yer alan değerler yöntem için aralık sayısının kullanıcı tarafından girilmeden yazılımın otomatik olarak ataması sonucu elde edilen sonuçlardır.

Tablo 5.8: Eşit genişlikli ayrıklaştırma yönteminin uygulanması sonucu elde edilen kesim noktası sayıları ve değerleri

Nitelik		Kesim Noktası Sayıları	Kesim Noktası Değerleri
ClumpThickness	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
CellSize	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
CellShape	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
MarginalAdhesion	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
EpithelialSize	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
BareNuclei	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
BlandChromatin	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
NormalNucleoli	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1
Mitoses	:	9	1.9, 2.8, 3.7, 4.6, 5.5, 6.4, 7.3, 8.2, 9.1

Tablo 5.8'de veri kümesi üzerinde eşit genişlikli ayrıklaştırma algoritmasının uygulanması sonucu elde edilen kesim noktası sayıları görülmektedir. ID3 ve eşit frekanslı algoritmalarında olduğu gibi her bir nitelik için birden fazla kesim noktası elde edilmiştir. Tablo 5.8'den de görüldüğü üzere tüm nitelikler için 9 adet kesim noktası yani 10 farklı kategori elde edilmiştir. Sonuçlar, yöntem için aralık değerinin yazılım tarafından otomatik olarak belirlenmesi sonucu elde edilmiştir.

5.2. KATEGORİK DEĞİŞKENLERE AİT BULGULAR

Tablo 5.9, 5.10, 5.11, 5.12, 5.13, 5.14, 5.15, 5.16 ve 5.17'de yer alan veriler tek bir niteliğe ait başlangıç değerlerini ve ayrıklaştırma sonrası elde edilen kategorik değerleri belirtmektedir. Nitelik sütunu niteliğin veri kümesinde yer alan gerçek değerini, diğer sütunlar da sırası ile 1RD, CADD, CAIM, Chi2, ChiMerge, ID3, Eşit frekanslı (EFD) ve Eşit genişlikli (EWD) ayrıklaştırma yöntemleri sonucu elde edilen kategorik değişkenlerin sayısal ifadelerini içermektedir. Tablolarda yer alan veriler, veri kümesinin genelini temsilen seçilen ilk 25 örneğe ait nitelik değerleridir.

Tablo 5.9: clumpThickness niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	1RD	CADD	CAIM	Chi2	ChiMerge	ID3	EFD	EWD
5	0	1	0	0	0	4	4	4
5	0	1	0	0	0	4	4	4
3	0	1	0	0	0	2	2	2
6	0	1	1	0	0	5	6	5
4	0	1	0	0	0	3	3	3
8	0	1	1	0	1	7	6	7
1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	2	1
2	0	1	0	0	0	1	2	1
4	0	1	0	0	0	3	3	3
1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	2	1
5	0	1	0	0	0	4	4	4
8	0	1	1	0	1	7	6	7
7	0	1	1	0	1	6	6	6
4	0	1	0	0	0	3	3	3
4	0	1	0	0	0	3	3	3
10	0	1	1	0	1	8	7	9
6	0	1	1	0	0	5	6	5
7	0	1	1	0	1	6	6	6
10	0	1	1	0	1	8	7	9
3	0	1	0	0	0	2	2	2
8	0	1	1	0	1	7	6	7
1	0	1	0	0	0	0	0	0
5	0	1	0	0	0	4	4	4

Tablo 5.9 incelendiğinde Chi2 ve 1RD algoritmalarının %100 oranda aynı sonucu verdiği görülmektedir. Bunun sonucu olarak her iki algoritma için tek bir kategori elde edilmiş ve nitelikler aynı kategori altında toplanmıştır. CAIM, CADD ve ChiMerge algoritmaları 0 ve 1 olmak üzere iki kategori oluşturmuşlardır. Bu algoritmalardan CAIM ve ChiMerge algoritmaları %90 oranında aynı sonucu vermiştir. ID3 algoritması 0 ile 8 arasında sayılar ile ifade edilen 9 kategori, EFD algoritması 0 ile 7 arasında

sayılar ile ifade edilen 8 kategori, EWD algoritması ise 0 ile 9 arasında sayılar ile ifade edilen 10 kategori oluşturmuştur.

Tablo 5.10: cellSize niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
1	0	1	0	0	0	0	0	0
4	0	1	1	1	1	3	3	3
1	0	1	0	0	0	0	0	0
8	0	1	1	1	1	7	4	7
1	0	1	0	0	0	0	0	0
10	0	1	1	1	1	9	5	9
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	1	1
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
3	0	1	0	0	0	2	3	2
7	0	1	1	1	1	6	4	6
4	0	1	1	1	1	3	3	3
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
7	0	1	1	1	1	6	4	6
1	0	1	0	0	0	0	0	0
3	0	1	0	0	0	2	3	2
5	0	1	1	1	1	4	4	4
1	0	1	0	0	0	0	0	0
4	0	1	1	1	1	3	3	3
1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	1	1

Tablol 5.10'da CADD, CAIM, Chi2 ve ChiMerge algoritmaları 0 ve 1 olmak üzere iki kategori oluşturmuşlardır. Bunun yanı sıra 1RD algoritması tek bir kategori, ID3 ve EWD algoritmaları 0 ile 9 arasında sayılar ile ifade edilen 10 kategori, EFD algoritması

0 ile 5 arasında sayılar ile ifade edilen 6 kategori oluşturmuştur. CAIM, Chi2 ve ChiMerge algoritmaları %100 oranında aynı sonuçları vermişlerdir. CADD algoritması ise bu algoritmalar ile yaklaşık olarak %30 oranında aynı sonucu vermiştir. Nitelikler üzerinde ID3 ve EWD algoritmaları uygulandığında %100 oranında aynı sonuçların alındığı görülmektedir.

Tablo 5.11: cellShape niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
1	0	1	0	0	0	0	0	0
4	0	1	1	1	1	3	3	3
1	0	1	0	0	0	0	0	0
8	0	1	1	1	1	7	4	7
1	0	1	0	0	0	0	0	0
10	0	1	1	1	1	8	5	9
1	0	1	0	0	0	0	0	0
2	0	1	0	1	1	1	2	1
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
3	0	1	0	1	1	2	3	2
5	0	1	1	1	1	4	3	4
6	0	1	1	1	1	5	3	5
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
7	0	1	1	1	1	6	4	6
1	0	1	0	0	0	0	0	0
2	0	1	0	1	1	1	2	1
5	0	1	1	1	1	4	3	4
1	0	1	0	0	0	0	0	0
5	0	1	1	1	1	4	3	4
1	0	1	0	0	0	0	0	0
3	0	1	0	1	1	2	3	2

Tablo 5.11'e göre cellShape niteliğine ait örnek değerlerine Chi2 ve ChiMerge ayrıklaştırma algoritmaları uygulandığında %100 oranında aynı sonuçları verdiği belirlenmiştir. 1RD algoritması tek bir, CADD, CAIM, Chi2 ve ChiMerge algoritmaları 0 ve 1 olmak üzere iki, ID3 algoritması 0 ile 8 arasında sayılar ile ifade edilen 9 kategori, EFD algoritması 0 ile 5 arasında sayılar ile ifade edilen 6 kategori, EWD algoritması 0 ile 9 arasında sayılar ile ifade edilen 10 kategori oluşturmuştur.

Tablo 5.12: marginalAthesion niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
1	0	1	0	0	0	0	0	0
5	0	1	1	0	1	4	4	4
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
3	0	1	0	0	1	2	2	2
8	0	1	1	0	1	6	4	7
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
3	0	1	0	0	1	2	2	2
10	0	1	1	0	1	8	5	9
4	0	1	1	0	1	3	4	3
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
6	0	1	1	0	1	5	4	5
1	0	1	0	0	0	0	0	0
10	0	1	1	0	1	8	5	9
3	0	1	0	0	1	2	2	2
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
4	0	1	1	0	1	3	4	3

Tablo 5.12’ye göre marginalAdheison niteliğine ait örnek değerlere CAIM ve ChiMerge ayrıklaştırma algoritmaları uygulandığında yaklaşık olarak %85 oranında, 1RD ve Chi2 algoritmaları uygulandığında da %100 aynı sonuçların elde edildiği belirlenmiştir. Tablo 5.12’de görüldüğü üzere 1Rd ve Chi2 algoritmaları tek bir, CADD, CAIM ve ChiMerge algoritmaları 0 ve 1 olmak üzere iki, ID3 algoritması 0 ile 8 arasında sayılar ile ifade edilen 9 kategori, EFD algoritması 0 ile 5 arasında sayılar ile ifade edilen 6 kategori, EWD algoritması 0 ile 9 arasında sayılar ile ifade edilen 10 kategori elde etmiştir.

Tablo 5.13: epithelialSize niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
2	0	1	0	0	0	1	1	1
7	0	1	1	1	1	6	5	6
2	0	1	0	0	0	1	1	1
3	0	1	1	1	1	2	2	2
2	0	1	0	0	0	1	1	1
7	0	1	1	1	1	6	5	6
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
1	0	1	0	0	0	0	1	0
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
7	0	1	1	1	1	6	5	6
6	0	1	1	1	1	5	4	5
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
4	0	1	1	1	1	3	4	3
2	0	1	0	0	0	1	1	1
5	0	1	1	1	1	4	4	4
6	0	1	1	1	1	5	4	5
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1
2	0	1	0	0	0	1	1	1

Tablo 5.13 göstermektedir ki epithelialSize niteliğine CAIM, Chi2, ChiMerge algoritmalarının uygulanması sonucu kendi aralarında, ID3 ve EWD algoritmalarının uygulanması sonucu da kendi aralarında %100 oranında aynı sonuçlar elde edilmiştir. Tablo 5.13'e göre 1RD algoritması tek bir, CADD, CAIM, ChiMerge, Chi2 algoritmaları 0 ve 1 olmak üzere iki, ID3 ve EWD algoritmaları 0 ile 8 arasında sayılar

ile ifade edilen 9 kategori, EFD algoritması da 0 ile 5 arasında sayılar ile ifade edilen 6 kategori oluşturmuştur.

Tablo 5.14: bareNuclei niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
1	1	1	0	0	0	0	0	0
10	1	1	1	1	1	8	4	9
2	1	1	0	1	1	1	1	1
4	1	1	1	1	1	3	2	3
1	1	1	0	0	0	0	0	0
10	1	1	1	1	1	8	4	9
10	1	1	1	1	1	8	4	9
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
3	1	1	1	1	1	2	2	2
9	1	1	1	1	1	7	3	8
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
10	1	1	1	1	1	8	4	9
1	1	1	0	0	0	0	0	0
10	1	1	1	1	1	8	4	9
7	1	1	1	1	1	5	3	6
1	1	1	0	0	0	0	0	0
Null	?	?	?	?	?	?	?	?
1	1	1	0	0	0	0	0	0
7	1	1	1	1	1	5	3	6

Tablo 5.14'e göre 1Rd ve CADD algoritmaları %100 olarak aynı sonuçları vermiştir. Bu oranda aynı sonuçları veren bir diğer algoritma çifti ise Chi2 ve ChiMerge algoritmalarıdır. Bunun yanı sıra CAIM algoritması da bu iki algoritma çifti ile yaklaşık

olarak %95 oranında aynı kategorik değerler elde etmiştir. Tablo 5.14 incelendiğinde 1RD, CADD, CAIM, Chi2 ve ChiMerge algoritmaları 1 ve 0 ile ifade edilmek üzere iki kategorik grup oluşturmuştur. bareNuclei niteliği için 1RD algoritması 0 ile 8 arasında sayılar ile ifade edilen 9 kategori, EFD algoritması 0 ile 4 arasında sayılar ile ifade edilen 5 kategori, EWD algoritmaları 0 ile 9 arasında sayılar ile ifade edilen 10 kategori meydana getirmiştir.

Tablo 5.15: blandChromatin niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
3	0	1	0	0	0	2	2	2
3	0	1	0	0	0	2	2	2
3	0	1	0	0	0	2	2	2
3	0	1	0	0	0	2	2	2
3	0	1	0	0	0	2	2	2
9	0	1	1	1	1	7	6	8
3	0	1	0	0	0	2	2	2
3	0	1	0	0	0	2	2	2
1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	1	1
3	0	1	0	0	0	2	2	2
2	0	1	0	0	0	1	1	1
4	0	1	1	1	1	3	3	3
5	0	1	1	1	1	4	5	4
4	0	1	1	1	1	3	3	3
2	0	1	0	0	0	1	1	1
3	0	1	0	0	0	2	2	2
4	0	1	1	1	1	3	3	3
3	0	1	0	0	0	2	2	2
5	0	1	1	1	1	4	5	4
7	0	1	1	1	1	6	5	6
2	0	1	0	0	0	1	1	1
7	0	1	1	1	1	6	5	6
3	0	1	0	0	0	2	2	2
3	0	1	0	0	0	2	2	2

Tablo 5.15'e göre CAIM, Chi2 ve ChiMerge algoritmaları %100 oranında aynı kategorik değerleri elde etmişlerdir. 1RD algoritması ile tek bir, CADD, CAIM, Chi2, ChiMerge algoritmaları 0 ve 1 olmak üzere iki farklı, ID3 algoritması ile 0 ve 7 arasında sayılar ile ifade edilen 8 farklı, EFD algoritması ile 0 ve 6 arasında sayılar ile ifade edilen 7 farklı, EWD algoritması ile 0 ve 9 arasında sayılar ile ifade edilen 10 farklı kategori oluşmuştur.

Tablo 5.16: normalNucleoli niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	2	1
1	0	1	0	0	0	0	0	0
7	0	1	1	1	1	6	3	6
1	0	1	0	0	0	0	0	0
7	0	1	1	1	1	6	3	6
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
4	0	1	1	1	1	3	2	3
5	0	1	1	1	1	4	2	4
3	0	1	1	1	1	2	2	2
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
4	0	1	1	1	1	3	2	3
10	0	1	1	1	1	9	4	9
1	0	1	0	0	0	0	0	0
3	0	1	1	1	1	2	2	2
1	0	1	0	0	0	0	0	0
6	0	1	1	1	1	5	3	5

Tablo 5.16'ya göre CAIM, Chi2 ve ChiMerge algoritmaları %100 oranında aynı kategorik değerleri elde etmişlerdir. Bu oranda aynı sonuçları veren bir diğer algoritma çiftleri ID3 ve EWD algoritmalarıdır. 1RD algoritması ile tek bir, CADD, CAIM, Chi2, ChiMerge algoritmaları 0 ve 1 olmak üzere iki farklı, ID3 ve EWD algoritması ile 0 ve 9 arasında sayılar ile ifade edilen 10 farklı, EFD algoritması ile 0 ve 4 arasında sayılar ile ifade edilen 5 farklı kategori elde edilmiştir.

Tablo 5.17: mitoses niteliğine ait 25 örneğin ayrıklaştırma öncesi gerçek ve ayrıklaştırma sonrası sayısal kategorik değerleri

Nitelik Değeri	**1RD**	**CADD**	**CAIM**	**Chi2**	**ChiMerge**	**ID3**	**EFD**	**EWD**
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
5	0	0	1	0	1	4	2	4
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
4	0	0	1	0	1	3	2	3
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	1	0	1	1	2	1
1	0	0	0	0	0	0	0	0
4	0	0	1	0	1	3	2	3
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0

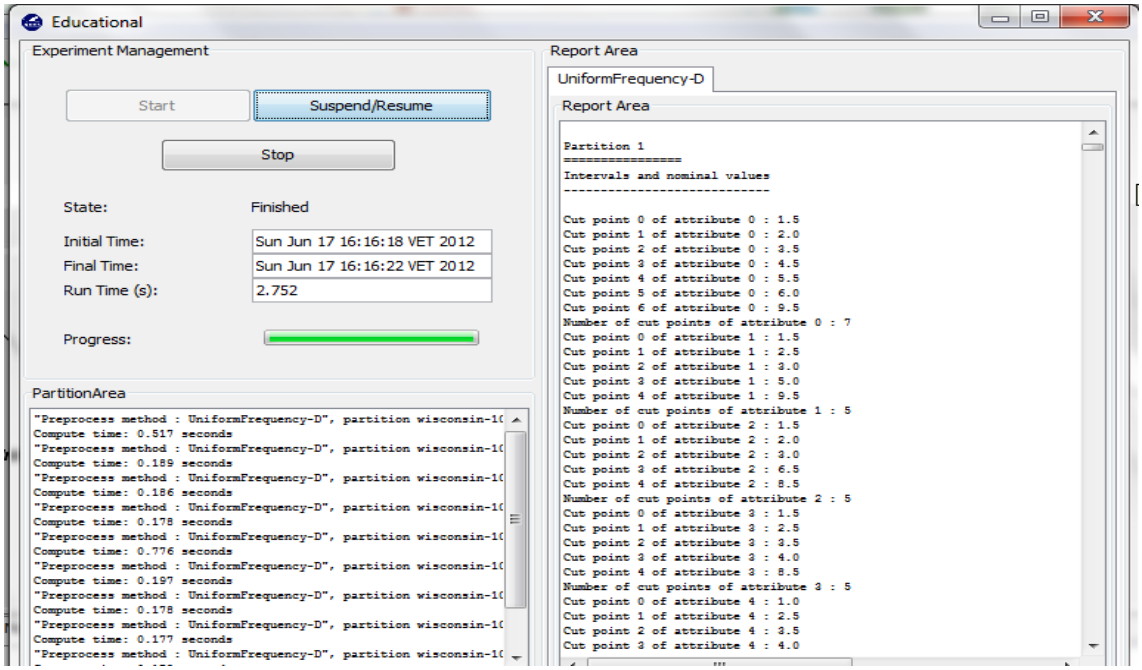
Tablo 5.17 incelendiğinde 1RD, CADD, Chi2 algoritmalarının tek bir kategorik değer elde etmiş olduğu dolayısıyla %100 oranında aynı sonuçları verdiği belirlenmiştir. Yine Tablo 5.17'ye göre CAIM ve ChiMerge algoritmalarının %100 uyumla çalıştığı ve aynı kategorik değişkenleri elde edip aynı niteliğe ait olan değerleri aynı kategori altında topladığı görülmektedir. Bir başka %100 oranla aynı sonuçları elde eden yöntem çiftlerinin ID3 ve EWD olmakla beraber bu algoritmalar 0 ile 9 arasında sayıların temsil ettiği 10 kategorik değişkeni oluşturmaktadır.

5.3. DİĞER BULGULAR

KEEL yazılımının kullanıcılar tarafından daha kolay öğrenilmesini amacı ile yazılım bünyesinde ayrıca "Eğitimsel Deneyler" fonksiyonu oluşturulmuştur. Bu fonsiyon kullanılarak hazırlanan deneyler sadece veri klasörü üretmekle kalmamakta yazılım arayüzünde de bazı sonuçları paylaşmaktadır. Ancak bu fonksiyonda uygulanabilecek veri ayrıklaştırma yöntemleri sınırlı sayıdadır.

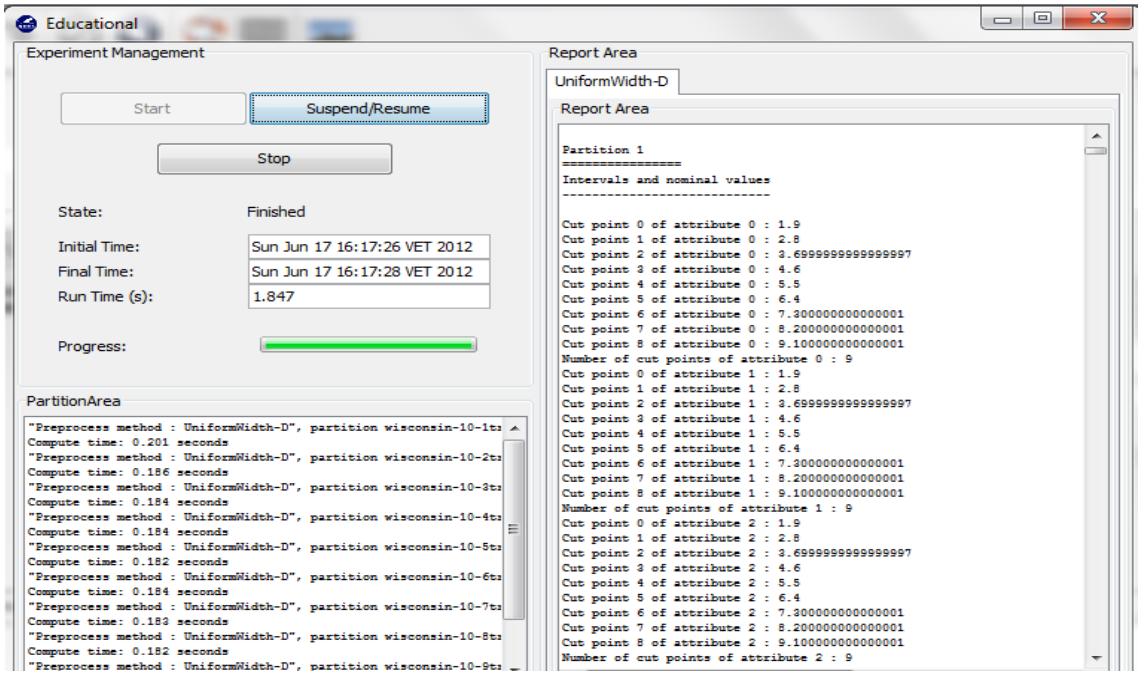
Bu fonkiyon altında tez kapsamında ele alınan ancak 2 yöntemin uygulanabilmesine izin verilmektedir. Bu yöntemler Eşit Genişlikli ve Eşit Frekanslı Ayrıklaştırma yöntemleridir. Eğitimsel deneyler fonksiyonunun kullanımı sonucu elde edilen bulgular bu başlık altında paylaşılmıştır.

Bu fonksiyon ile elde edilen bulgular, daha önce tablolar halinde ifade edilmiş kategorik değişken sayısı ve değerleri olmak ile beraber açık olarak hem toplam süre hem de kullanılan önişleme yöntemi ve veri kümesi bilgisini kapsamaktadır.



Şekil 5.1: Eşit Frekanslı Ayrıklaştırma Yöntemi için eğitimsel deney fonksiyonu çıktısı

Şekil 5.1'de görüldüğü üzere yöntem için sonuçlar 2.752 saniyede elde edilmiştir. Görselin sağ kısmında da bu yöntem sonucunda her bir nitelik için elde edilen kesim noktaları sayıları ile bu noktaların değerleri verilmiştir. Buna göre Tablo 5.7'de verilen değerlerin yazılım arayüzündeki listelenmiş hali görülmektedir.



Şekil 5.2: Eşit Genişlikli Ayrıklaştırma Yöntemi için eğitimsel deney fonksiyonu çıktısı

Şekil 5.2’de de Eşit Genişlikli Ayrıklaştırma yöntemi için yazılım arayüzünde listelenen sonuçlar görünmektedir. Buna göre bu yöntem için yazılım 1.847 saniyede işlem yapmış yine Tablo 5.8’de belirtilen değerler ile aynı olmak üzere görselin sağ kısmında yer alan kesim noktaları sayıları ve değerleri elde edilmiştir.

6.TARTIŞMA VE SONUÇ

‘Elde Edilen Kategorik Değişkenlere Ait Bulgular’ ve ‘Kesim Noktası Sayılarına Ait Bulgular’ başlıkları altında belirtilen sonuçlar için yöntemlere göre oluşturulan kesim noktaları sayıları, bu noktaların oluşturduğu aralık değerleri, bu aralığa düşen nitelik değerleri ve aynı nitelik değerine sahip örneklerin kategorik değişken değerleri incelendiğinde elde edilen bulguların tutarlı olduğu belirlenmiştir.

Bazı nitelikler için özellikle CAIM, Chi2 ve ChiMerge; ID3, Eşit Frekanslı (EFD) ve Eşit Genişlikli (EWD) Ayrıklaştırma algoritmalarının sayılan bu gruplar dahilinde %100 uyumlu çalıştıkları, nitelikler için atanmış kategorik değişkenlerin birebir aynı veya birbirine çok yakın olduğu sonucuna ulaşılmıştır. CADD algoritmasının CAIM, Chi2, ChiMerge algoritmaları ile çok yakın olmasa da benzer değerler atadığı belirlenmiştir.

Chi2 ve ChiMerge algoritmaları ya da EFD ve EWD algoritmaları arasında benzerliklerin bulunması sonucunu normal karşılamak gerekir. Bunun nedeni olarak, bu iki grup algoritmanın kendi içlerinde aynı veri ayrıklaştırma yöntemi kategorisinde bulunmaları gösterilebilir.

Yazılım aracı ile deney tasarımı yapılırken eşit genişlikli ve eşit frekanslı ayrıklaştırma yöntemleri için aralık sayısı girilmemiş; yazılımın otomatik olarak aralık sayısı ataması beklenmiştir. Dolayısıyla bu iki yöntem için CADD, CAIM, Chi2, ChiMerge algoritmalarına benzer iki veya üç aralık oluşturması istenseydi, elde edilen bulgularda bu iki algoritmanın da sözü geçen dört algoritma ile ne kadar benzer sonuçlar verebileceği araştırılabilirdi.

Yine bulgular göstermektedir ki; yöntemlerin uygulanması sonrası elde edilen kategorik sonuçlar yukarıda belirtildiği gibi bazı nitelik değerleri için uyumluluk gösterirken, veri

kümesine ait tüm nitelikler için karşılaştırma yapıldığında sonuçlar bütünüyle birbiri ile aynı değildir. Bu sonuçların birbiri ile olan tutarlılığının bazı nitelikler için atanan kategorik değişkenlerin sayısı nedeniyle bozulduğu görülmüştür. Bunun sonucu olarak, yakın kategorik değişkenlerin atandığı belirlense de birebir tutarlılık söz konusu değildir. Belirtilen bozulmaların kategori sayısının fazla olduğu yöntemlerde daha fazla meydana geldiği söylenebilir.

Diğer bulgular başlığı altında yer verilen bulgular incelendiğinde, kullanılan yazılımın Eşit Genişlikli Ayrıklaştırma yöntemi için sonuçları daha kısa sürede elde ettiği sonucuna ulaşılmıştır.

Yapılan literatür taramasında incelenen çalışmaların bazılarında kesim noktası sayısının dolayısı ile çok daha fazla sayıda kategorik değişken elde etmenin veri kaybını azaltacağı, dolayısı ile daha iyi bir ayrıklaştırma sonucu vereceği belirtilmiştir. Ancak bu çalışmanın devamı niteliğinde, en iyi yöntemin araştırılabileceği bir çalışma kapsamında, yöntemlerin kalite kriterlerini sağlayıp sağlamadıkları ölçümlenerek elde edilecek bulgular ile daha doğru sonuçlar elde edilebilir.

Tez çalışmasının başında belirtilen amaçlar dahilinde tüm ayrıklaştırma yöntemlerini kapsayacak şekilde, bahsedilen benzerliklerin yüzdelik bir oran ile tek bir paydada toplanarak ifade edilebilmesi mümkün olmamıştır.

Veri ayrıklaştırma için kullanılacak çok sayıda yöntemin varlığı söz konusudur. Ancak uygulama için kullanılacak yazılımlar incelendiğinde bu yazılımların birçoğunda ChiMerge, Chi2, Eşit Genişlikli ve Eşit Frekanslı ayrıklaştırma yöntemlerinin yer aldığı gözlemlenmiş olup, ayrıklaştırma için bu yöntemlerin daha sık kullanıldığı sonucuna ulaşılabilir.

Ayrıklaştırma algoritmaların birbirlerine göre avantaj ve dezavantajlarının olduğu bilinmekte, bu avantaj ve dezavantajlar gözönünde bulundurularak yeni birçok yöntem geliştirilmektedir. Daha önce de belirtildiği üzere; örneğin eşit genişlikli ayrıklaştırma yönteminin Boulle'ye göre (2005) yalın olması diğer yöntemlerin arasında bu yöntemi cazip hale getirse de, Wong ve diğ. (1987), aralık genişliklerinin belirlenerek eşit

aralıklara bölme işlemi doğru yapılamadığı taktirde bu yöntem ayrıklaştırma işlemi sonrasında büyük bir veri kaybına neden olabileceği yönünde görüş bildirmişlerdir. Başka açıdan Chi2 gibi bazı yöntemler kullanıcıdan parametre girmesini beklerken ChiMerge gibi bazı yöntemler de parametreleri kendileri belirlemektedir. Bu durumda kullanıcının gireceği parametre değerinin hatalı olması elde edilecek sonuçlar açısından sorun teşkil edebileceği söylenebilir.

Bu tez çalışmasının sonucu olarak her ayrıklaştırma yöntemi her veri kümesi için uygun mudur sorusu da akıllara gelmiştir. Örneğin çalışma kapsamında kullanılan veri kümesi için 1RD algoritması uygulandığında tek bir değişken elde edilmiş tüm değerler aynı kategori altında toplanmıştır. Bunun yanı sıra ID3 algoritması sonucunda da 5 ile 10 arasında kategorik değişken elde edilmiştir. Bu noktada hangi yöntemin seçilmesi gerektiği de yeni bir çalışma alanı yaratmaktadır. Yapılabilecek doğruluk analizleri bahsi geçen çalışma için yol gösterici olacaktır.

Tüm bunların yanı sıra uygulamayı gerçekleştirebilmek için araştırılan yazılımların genelde veri madenciliği yöntemlerine yönelik olduğu söylenebilir. Hem ayrıklaştırma yöntemlerinin uygulanabilirliğini kolaylaştıracak hem de görsel açıdan daha zengin, kullanımı kolay ve en önemlisi raporlama yeteneği güçlü yazılım araçlarına ihtiyaç bulunmaktadır.

KAYNAKLAR

AKPINAR, H., 2000, Veri Tabanlarında Bilgi Keşfi ve Veri MAdenciliği, *İ.Ü. İşletme Fakültesi Dergisi*, 29 (1), 1-22.

ALCALÁ-FDEZ, J., FERNANDEZ, A., LUENGO J., DERRAC, J., GARCIA, S., SÁNCHEZ, L., HERRERA, F., 2011, KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework, *Journal of Multiple-Valued Logic and Soft Computing*, 17 (2-3), 255-287.

ALCALÁ-FDEZ, J., SÁNCHEZ, L., GARCIA, S., JESUS, M.J., VENTURA, S., GARRELL, J.M, OTERO, J., ROMERO, C., BACARDIT, J., RIVAS, V.M., FERNÁNDEZ, J.C., HERRERA, F., 2009, KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems, *Journal of Multiple-Valued Logic and Soft Computing*, 13 (3), 307-318.

BAKAR, A.A., OTHMAN, Z.A., SHUIB, N.L.M., 2009, Building A New Taxonomy For Data Discretization Techniques, 2nd Conference on Data Mining and Optimization, Malezya.

BANDYOPADHYAY, S., PAL, S. K., 2007, *Classification and Learning Using Genetic Algorithms Applications in Bioinformatics and Web Intelligence*, Springer-Verlag Berlin Heidelberg, 978-3-540-49606-9.

BAO, Y., TSUCHIYA, E., ISHII, N., 2005, Classification by Instance-Based Learning Algorithm, *Lecture Notes in Computer Science (LNCS)*, 3578, 133-140.

BIBA, M., ESPOSITO, F., FERILLI, S., MAURO, N.D., BASILE, T.M.A., 2007, Unsupervised Discretization Using Kernel Density Estimation, *Proceedings of the 20th international joint conference on Artifical intelligence (IJCAI), 6-12 Ocak 2007 Hyderabad India, SanFrancisco*, CA, USA:Morgan Kaufmann Publishers Inc., 696-701.

BOULLE, M., 2005, Optimal Bin Number For Equal Frequency Discretizations in Supervised Learning, *Intelligent Data Analysis*, 9 (2), 175-188.

CERQUIDES, J., LOPEZ DE MANTARASE, R., 1997, Proposal and Empirical Comparision of a Parallelizable Distance-Based Discretization Method, *III. International Conference on Knowledge Discovery and Data Mining (KDDM97), 14-17 Ağustos 1997, California*, AAAI Press, 139-142.

CHANG, Y.C., LAI, P.C., LEE, M.T., 2007, An Integrated Approach For Operational Knowledge Acquisition of Refuse Incinerators, Expert Systems with Applications, 33, 413-419.

CHAKRABARTI, S., COX, E., FRANK, E., GTING, R.H., HAN, J., JIANG, X., KAMBER, M., LIGHTSTONE, S.S., NADEAU, T.P, NEAPOLITAN, R.E., PYLE, D., REFAAT, M., SCHNEIDER, M., TEOREY, T.J., WITTEN, I.H., 2008, *Data Mining: Know It All*, Morgan Kaufmann Publishers Inc., San Francisco, CA, 978-0-12-374629-0.

CHAOKUN, Y., JIANPING, L., ENMING, D., 2011, A Discretization Algorithm Based on Clustering and CAIR Criterion, İçinde: Ding, Y., Wang, H., Xiong, N., Hao, K., Wang, L. (ed.), *Proceedings of The Seventh International Conference on Natural Computation (ICNC-2011), 26-28 Temmuz 2011 Shangai, China,* IEEE Conference Publications, 1424-1429.

CHEN, J.S., CHENG, C.H., 2008, Extracting Classification Rule of Software Diagnosis Using Modified MEPA, *Expert Systems with Applications*, 34, 411-418.

CHEN, M.S., HAN, J., YU, P.S., 1996, Data Mining: An Overview From A Database Perspective, *IEEE Transactions on Knowledge and Data Engineering*, 8 (6), 866-883.

CHING, J.Y., WONG, A.K.C., CHAN, K.C.C., 1995, Class-Dependent Discretization for Inductive Learning from Continuous and Mixed-Mode Data, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 17 (7), 641-651.

DAS, K., VYAS, O.P., 2010, A Suitability Study of Discretization Methods for Associative Classifiers, *International journal of Computer Applications*, 5 (10), 46-51.

DOUGHERTY, C., KOHAVI, R., SAHAMI, M., 1995, Supervised and Unsupervised Discretization of Continuous Features, İçinde: Prieditis, A., Russell, S.J. (ed), *Machine Learning: Proceedings of the Twelfth International Conference on Machine Learning, 9-12 Temmuz 1995 Tahoe City, California, USA*, Morgan Kaufmann, 194-202.

DUNHAM, M.H., 2003, *Data Mining Introductory and Advanced Topics*, Pearson Education Inc., USA, 0-13-088892-3.

FAYYAD, U., PIATETSKY-SHAPIRO, G., SMYTH, P., 1996, Knowledge Discovery And Data Mining: Towards a Unifying Framework, *II. International Conference on Knowledge Discovery and Data Mining (KDDM96), 2-4 Ağustos 1996, Portland, Oregon,* AAAI Press, 82-88.

FAYYAD, U.M., 1996, Data Mining and Knowledge Discovery: Making Sense Out of Data, *IEEE Expert: Intelligent Systems and Their Applications*, 11 (5), 20-25.

FRAWLEY, W.J., PIATETSY-SHAPIRO, G., MATHEUS, C.J., 1992, Knowledge Discovery in Databases:An Overview, *AI Magazines*, 13 (3), 57-70.

FREITAS, A.A., 2002, *Data Mining and Knowledge Discovery with Evolutionary Algorithms*, Springer, Germany, 3-540-43331-7.

FRIEDMAN, J.H., 1997, Data *Mining and Statistics: What's the Connection?* [online], Standford University, http://www-stat.stanford.edu/~jhf/ftp/dm-stat.pdf [Ziyaret Tarihi: 8 Mayıs 2012].

GORUNESCU, F., 2011, *Data Mining Concepts, Model and Techniques*, Springer, Verlag-Berlin-Heidelberg, 978-3-642-19720-8.

GUNN, S.R., BROWN, M., BOSSLEY, K.M., 1997, Network Performance Assesments for Neurofuzzy Data Modelling, *Lecture Notes in Computer Science*, 1280, 313-323.

HAN, J., KAMBER , M., 2001, *Data Mining Concepts and Techniques*, Morgan Kaufmann Publishers, Massachusetts**,** USA, 978-0-12-381479-1.

HAND, D., MANNILA, H., SYMTH, P., 2001, *Principles of Data Mining*, Massachusetts Institute of Technology Press, USA, 978-0-262-08290-7.

HOLTE, R.C., 1993, Very Simple Classification Rules Perform Well on Most Commonly Used Datasets, *Machine Learning*, 11, 63-91.

HOLTE, R.C., ACKER, L., PORTER, B.W., 1989, Concept Learnning and the Problem of Small Disjunts, *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, 20-26 Ağustos 1989, Detroit, MI,* Morgan Kaufman, 813-818.

HSU, C.N., HUANG, H.J., WONG, T.T., 2003, Implications of the Dirichlet Assumption for Discretization of Continuous Variables in Naïve Bayesian Classifiers, *Machine Learning*, 53, 235-263.

INMON, W.H., 1992, *Building The Data Warehouse*, John Wiley & Sons, Inc., New York, 9780471569602.

INMON, W.H., 2000, *What is a Data Warehouse?* [online], Aalborg Universitet, https://www.business.auc.dk/oekostyr/file/What_is_a_Data_Warehouse.pdf [Ziyaret Tarihi: 9 Mayıs 2012].

INMON, W.H., KELLY, C., 1994, The Twelve Rules of Data Warehouse for A Client/Server World, *Data Management Review*, 4 (5), 6-16.

ISMAIL, M.K., 2003, *An Imperical Investigation of The Impact of Discretization On Common Data Distrubitions*, Yüksek Lisans Tezi, RMIT University.

JIN, R., BREITBART, Y., MUOH, C., 2007, Data Discretization Unification, *Seventh IEEE International Conference on Data Mining (ICDM), 28-31 Ekim 2007 Omaha Nebreska,* IEEE Conference Publications, 183-192.

KANTARDZIC, M., 2011, *Data Mining Concepts, Models, Methods and Algorithms*, John Wiley & Sons, Inc., New Jersey, 978-0-470-89045-5.

KAVLI, T., 1993, ASMO-Dan Algorithm for Adaptive Spline Modelling of Observation Data, *International Journal of Control*, 58 (4), 947-967.

KERBER, R., 1992, ChiMerge: Discretization of Numeric Attributes. *The Tenth National Conference on Artificial Intelligence American Association for Artificial Intelligence (AAAI-92), 12-16 Temmuz 1992, California*, 123-128.

KHAN, A., REVETT, K., 2004, Data Mining the PIMA Dataset Using Rough Set Theory with a Special Emphasis on Rule Reduction, *Proceedings of INMIC 8th International Multitopic Conference, 24-26 Aralık 2004 Lahore Pakistan*, IEEE Conference Publications, 334-339.

KIMBALL, R., ROSS, M., 2002, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, John Wiley & Sons, Inc, New York, USA, 9780471200246.

KURGAN, A.L., CIOS, K.J., 2003, Fast Class-Attribute Interdependence Maximization (CAIM) Discretization Algorithm, İçinde: Wani, M.A., Cios, K.J., Hafeez, K. (ed.), *Proceedings of International Conference on Machine Learning and Applications, 23-24 Haziran 2003, Los Angeles California, USA*, CSRA Press, 30-36.

KURGAN, A.L., CIOS, K.J., 2004, CAIM Discretization Algorithm, *IEEE Transactions on Knowledge And Data Engineering*, 16 (2), 145-153.

KOTSIANTIS, S., KANELLOPOULOS, D., 2006, Discretization Techniques:A Recent Survey, *GESTS International Transactions on Computer Science and Engineering*, 32 (2), 47-58.

KOYUNCUGİL, A. S., 2007, *Borsa Şirketlerinin Sektörel Risk Profillerinin Veri Madenciliği ile Belirlenmesi* [online], Sermaye Piyasası Kurulu Raporu, T.C. Başbakanlık Sermaye Piyasası Kurulu, http://www.spk.gov.tr/yayin.aspx?type=yay03 [Ziyaret Tarihi: 25 Aralık 2011].

LEE, C.H., 2005, Discretizing Continuous Attributes Using Information Technology, İçinde: Yolum, P., Güngör, T., Gürgen, F., Özturan, C. (eds.), *Proceedings of 20th International Symposium on Computer and Information Sciences (ISCIS), Ekim 2004, Turkey*, Springer, 493-502.

LI, M., DENG, S., FENG, S., FAN, J., 2011, An Effective Discretization Based on Class-Attribute Coherence Maimization, *Pattern Recognition Letters*, 32, 1962-1973.

LIANG, X., XUE, C., HUANG, M., 2010, *Improved Apriori Algorithm for Mining Rules of Many Diseases*, Springer-Verlag Berlin Heidelberg, 978-3-642-16387-6.

LIU, H., HUSSAIN, F., TAN, C.L., DASH, L., 2002, Discretization: An Enabling Technique, *Data Mining and Knowledge Discovery*, 6 (4), 393-423.

LIU, H., SETINO, R., 1997, Feature Selection via Discretization, *IEEE Transactions on Knowledge And Data Engineering*, 9(4), 642-645.

LIU, H., SETİNO, R., 1995, Chi2:Feature Selection and Discretization of Numeric Attributes, *Proceedings of The IEEE 7th Internaional Conference on Tools with Artificial Intelligence, 5-8 Kasım 1995, Herndon Virginia*, IEEE Conference Publications, 388-391.

MAIMON, O., ROKACH, L., 2005, *The Data Mining and Knowledge Discovery Handbook*, Springer, USA, 978-0-387-24435-8.

MEHTA, S., PARTHASARATHY S., YANG, H., 2005, Toward Unsupervised Correlation Preserving Discretization, *IEEE Transactions On Knowledge And Data Engineering*, 17 (9), 1174-1185.

MITCHELL, T.M., 1997, *Machine Learning*, MIT press and the McGraw-Hill Companies Inc., Singapore, 0-07-042807-7.

MULLER, H., FREYTAG, J. C., 2003, *Problems, Methods and Challenges in Comprehensive Data Cleansing* [online], Humboldt Universitat, Berlin, http://www.dbis.informatik.hu-berlin.de/fileadmin/research/papers/techreports/2003-hub_ib_164-mueller.pdf [Ziyaret Tarihi: 15 Nisan 2012].

OLSON, D.L., DELEN, D., 2008, *Advanced Data Mining Techniques*, Springer, Verlag Berlin Heidelberg, 978-3-540-76916-3.

ÖZKAN, Y., 2008, *Veri Madenciliği Yöntemleri*, Papatya Yayıncılık, İstanbul, 978-975-6797-82-2.

ROLDAN, M. C., 2010, *Pentaho 3.2. Data Integration Beginner's Guide*, Packt Publishing, Birmingham, UK, 978-1-847199-54-6.

SHMUELI, G., PATEL, N. R., BRUCE, P. C., 2010, *Data Mining for Business Intelligence Concepts, Techniques and Applications in Mıcrosoft Office Excel with XLMiner*, John Wiley & Sons, Inc., New Jersey, 978-0470526828.

SU, C.T., HSU, J.H., 2005, An Extended Chi2 Algortihm for Discretization of Real Value Attributes, *IEEE Transactions on Knowledge And Data Engineering*, 17 (3), 437-441.

SULLIVAN, R., 2011, *Introduction to Data Mining for the Life Sciences*, Springer, New York, USA, 978-1-58829-942-0.

QUINLAN, J. R., 1993, *C4.5: Programs For Machine Learning*, Morgan Kaufmann Pub., USA, 1-55860-238-0.

TAKÇI, H., SOĞUKPINAR, İ., 2002, Saldırı Tespitinde En Yakın K-Komşu Uygulaması, VIII. Türkiyede Internet Konferansları, 19-21 Aralık 2002, İstanbul.

TAN, P. N., STEINBACH, M., KUMAR, V., 2006, *Introduction to Data Mining*, Pearson Education Inc., Boston, 0-321-32136-7.

TANASA, D., TROUSSE, B., 2004, Advanced Data Preprocessing For Intersites Web Usage Mining, *IEEE Intelligent Systems*, 19 (2), 59-65.

TAY, F.E.H., SHEN, L., 2002, A Modified Chi2 Algorithm for Discretization, *IEEE Transactions on Knowledge And Data Engineering*, 14 (3), 666-670.

TSAI, C.J., LEE, C.I., YANG, W.P., 2008, A discretization algorithm based on Class Attribute Contingency Coefficient, *Information Sciences*, 178 (3), 714-731.

Türk Dil Kurumu (TDK), 2011, *Büyük Türkçe Sözlük* [online], http://www.tdk.gov.tr/ [Ziyaret Tarihi: 19 Şubat 2012].

TÜRKOĞLU, İ., 2007, Karar Ağaçları Ve Fraktal Analiz Kullanarak Histopatolojik İmgelerin Sınıflandırılması, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 22 (4), 753-758.

WONG, A.K.C., CHIU, D.K.Y., 1987, Synthesizing Statistical Knowledge from Incomplete Mixed-Mode Data, *IEEE Transactions On Pattern Analysis And Machine Intelligence*, PAMI-9 (6), 796-805.

YALÇINKAYA, İ., 2012, *Bilgi Yönetiminin Örgütsel Etkileri*, İçinde: GÜLSEÇEN, S. (ed.), Bilgi ve Bilginin Yönetimi, Papatya Yayıncılık, İstanbul, 37-50.

YANG, Y., WEBB, G.I., 2005, *Discretization for Data Mining*, İçinde: Wang, J. (ed.), Encyclopedia of Data Warehousing and Mining, IGI-Global Disseminator of Knowledge, USA, 392-396.

YANG, Y., WEBB, G.I., 2002, A Comparative Study of Discretization Methods for Naïve-Bayes Classifiers, İçinde: Yamaguchi, T., Hoffmann, A., Motoda, H., Compton, P. (ed.), *Proceedings of the 2002 Pacific Rim Knowledge Acquisition Workshop (PKAW'02) Tokyo, Japan,* Tokyo: Japanese Society for Artificial Intelligence, 159-173.

ÖZGEÇMİŞ

Fatma Önay KOÇOĞLU, 20.07.1984 tarihinde İstanbul'da doğmuştur. 2002 yılında Fatih Pertevniyal Anadolu Lisesi'nde lise eğitimini, sonrasında da İstanbul Üniversitesi Fen Fakültesi Matematik Bölümü'nde lisans eğitimini tamamlamıştır. 2009 yılında İstanbul Üniversitesi Enformatik Bölümü'nde yüksek lisans eğitimine başlamıştır. 2010 yılından bu yana İstanbul Üniversitesi Enformatik Bölümü'nde Araştırma Görevlisi olarak çalışmaktadır. İyi derecede İngilizce ve ileri başlangıç seviyesinde Fransızca bilmektedir. Özellikle Veri madenciliği, veri tabanı sistemleri, web tasarım, geometri konularına ilgi duymaktadır.

Yayınları:

Gülseçen, S., Kartal Karataş, E., **Koçoglu, F. Ö.** (2012). Can GeoGebra Make Easier The Understanding Of Cartesian Co-Ordinates? A Quantitative Study In Turkey. 3rd International Conference on New Trends in Education and Their Implications, Antalya, Baskıda.

Ayvaz Reis, Z., Özdemir, Ş., Ugraş, T., **Koçoglu, F. Ö.** (2011). How Much the Concept of E-Government and Its Applications Came into Our Life: The Case of Informatics Department, Istanbul University. World Conference on Information Technology 2011, Antalya, Baskıda.

Gürsul, F., Yiğitbaşı, İ., **Koçoğlu, F. Ö.** (2011). Üniversite Öğrencilerinin Askerlik Süresine ve Askere Alma Yöntemlerine İlişkin Görüşleri. Uluslararası Yükseköğretim Kongresi: Yeni Yönelişler ve Sorunlar, 27 - 29 Mayıs 2011, İstanbul, Baskıda.

Saraç, A.E., **Koçoğlu, F.Ö.,** Ayvaz Reis, Z. (2011). Web Tabanlı Eğitimde İçerik Tasarımı. Akademik Bilişim'11 - XIII. Akademik Bilişim Konferansı Bildirileri, 1-5 Şubat 2011 İnönü Üniversitesi, Malatya, Baskıda.